INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY

HYDERABAD

CSE481: Optimization Methods

Professor : CV Jawahar
Notes By : Sai Manaswini Reddy I

2021

Optimization : the action of making the best or most effective use of a situation or resource

How do we achieve or meet our goals/expectations in the best possible form is what optimization looks at. We are interested in setting up A solution or finding a solution that is best within the constraint.

Objectives in optimization :

- Learn to state optimization problems and formulate known optimization problems formally.
- There are a number of solution schemes depending on the nature of the problem. Identifying the problem class, learning some of the popular classes of solution schemes.
- Appreciate the theoretical aspects of optimality, complexity in solving them, and things that can be guaranteed.

Classes in optimization :

- Linear vs Non-linear
- Convex vs Non-convex
- Discrete vs Continuous
- Constrained vs Unconstrained

Note: You should aim to develop skills that can help you to place a problem in an appropriate class.

Linear programming :

Objective function :

An objective function is part of a linear programming optimization strategy, which finds the minimum or maximum of a linear function.

Feasible region :

the set of points my solution is acceptable to me/ feasible/ constraints are satisfied/ solution space. Any point in this region is a feasible point.

Optimal solution :The solution of a linear programming problem reduces to finding the optimum value (largest or smallest, depending on the problem) of the linear expression (called the objective function)

The objective function and the constraints placed upon the problem must be deterministic and able to be expressed in linear form.

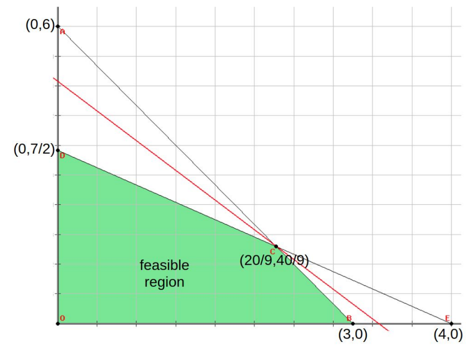
Example :

$$\textbf{Maximize } \mathbf{z, z = 3x_1 + 2x_2} \text{ [objective function]}$$

Subject to [constraints]

$$2x_1 + x_2 \leq 6$$
$$7x_1 + 8x_2 \leq 28$$
$$x_1, x_2 \geq 0 ;\ x_1 \in \mathbb{R} ; x_2 \in \mathbb{R}$$

Plot (using php simplex) :



If we do not have any constraints to the objective function over x_1,x_2 then the solution will end up being infinite. This is a constraint optimization problem. And we need the optimal values of x_1 and x_2 where z is maximum are x_1^* and x_2^* with an optimal value of z^*.

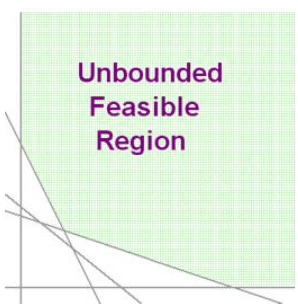
Shaded part is the feasible region.

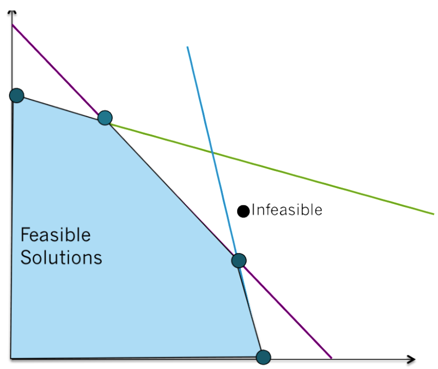
Where will $3x_1+2x_2$ get maximized and how ?

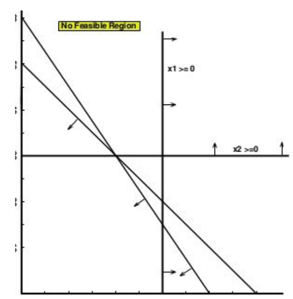
- Let us consider a point inside the polygon. K, this cannot be the best value because I can increase A&B. In such a way that my objective will increase. I can move. So my objective here is 3 X_1+2X_2. I can move from a B point in some direction such that 3X_1 plus 2X_2 will increase. One of the directions, I don't know which direction. I'm sure you can guess what will be the directions in one of the directions I can move from the AB such that 3X1_1+2X_2 is maximized.
- On looking at the graph we can say that the optimals solution cannot be inside the polygon, it has to be on the boundary or optimals solution might be one of the vertices.
- Solution will be one among the four vertices.
- We do brute force search on the four vertices.

We can draw parallel lines $3x_1+2x_2 = c_i$, i ∊ {1,2,3,...}. Starting from $c_1 = 0$, we keep on increasing c_i until we reach a point where no point on the line will satisfy the constraints. Line does not pass through the shaded region. Best one is the one passing through (20/9 , 40/9).

A particular optimal value z^* can have multiple x_1^*, x_2^*. There may exist a situation such that a feasible region might not exist at all.







Compact representation of LP (Standard Form) :

$$\text{Maximize } z = c^T \cdot x$$
$$\text{Subject to } A.x \leq b$$
$$x \geq 0$$

X is a vector, A is a matrix.

Example :

For the previous example :

$$c = [3,2]^T ; b = [6,28]$$
$$A = \begin{bmatrix} 2 & 1 \\ 7 & 8 \end{bmatrix}$$

Feasible region is always convex :

Convex Region :

A region is convex if a line joining any two points in this region also lies in this region then this region is convex. Set of all the points in this region is a convex set.

Note : if 2 sets are convex then their intersection is also convex.

Proof :

Let us consider 2 points x_1,x_2 from the feasible region.

Line joining them is represented by $px_1+(1-p)x_2$, $p \in (0,1)$

$x_1,x_2 \in$ feasible region $\Longrightarrow$

- $A.x_1 \leq b$ (1)
- $A.x_2 \leq b$ (2)

$$p(\text{eq-}(1)) + (1-p)(\text{eq-}(2)) \Longrightarrow p(A.x_1)+(1-p)(A.x_2) \leq p(b)+(1-p)(b)$$

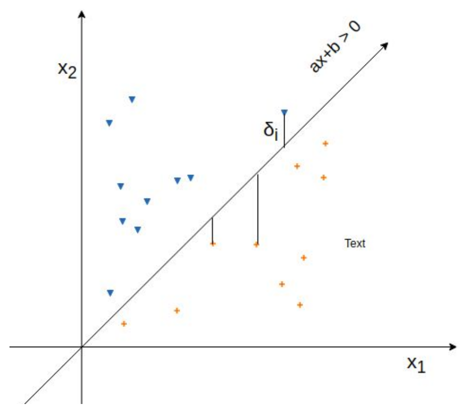
$$A.(px_1+(1-p)x_2) \leq b \ \forall\, p \in (0,1)$$

$\Longrightarrow$ line joining x_1 and x_2 also lies in the feasible region.

Classification as LP problem :

We have a two sets of samples

- $(x_i,y_i)^+$, i = 1,2,3,..n+
- $(x_i,y_i)^-$, i = 1,2,3,..n+



We want y-ax+b>0 for all the positive samples and <0 for all the negative samples.

Pattern Classification as LP :

Objective function :

$$\text{Maximize } \delta$$

Subject to :

- $y_i^+ \geq ax_i^+ + b + \delta \quad \forall\, i \in \{1,2,3,....n+\}$
- $y_i^- \leq ax_i^- + b - \delta \quad \forall\, i \in \{1,2,3,....n-\}$

We want the samples to be as far as possible from the line. So, we want δ to be maximized. The nearest samples should be as far as possible. All samples should be atleast δ away (constraint 1)

δ : minimum of all the possible distances between samples and the line.

We can also formulate this without considering δ.

New objective function : Maximize 1 ; without changing the constraints.

Exception :

- If the samples are not linearly separable. Then, there is no way we can get an a,b that satisfies the constraints. Feasible region is empty.

Note :

- We can use δ_1 and δ_2 instead of δ, and maximize $p\delta_1+q\delta_2$ [weightage to δ_1 and δ_2 based on our requirement]
- This is similar to SVM. There we use perpendicular distance, here we use distances measured along y axis.

Standard form :

Objective function = δ = 0.a + 0.b + 1.δ

$$\Longrightarrow c = [0\ 0\ 1]$$

$$x = [a\ b\ \delta]$$

Subjective :

- $ax_i^+ + b + \delta \leq y_i^+$
- $ax_i^- + b - \delta \geq y_i^- \Rightarrow -ax_i^- - b + \delta \leq y_i^-$

$$A = \begin{bmatrix} x_1^+ & x_2^+ & . & . & x_{n+}^+ & -x_1^- & -x_2^- & . & . & -x_{n-}^- \\ 1 & 1 & . & . & 1 & -1 & -1 & . & . & -1 \\ 1 & 1 & . & . & 1 & 1 & 1 & . & . & 1 \end{bmatrix}$$

$$b = \begin{bmatrix} y_1^+ & y_2^+ & . & . & y_{n+}^+ & -y_1^- & -y_2^- & . & . & -y_{n-}^- \end{bmatrix}$$

Line fitting (Regression) as LP :

We have a set of samples

$$(x_i, y_i)\ ,\ i = 1,2,3,..n$$

$$y = ax_i + b$$

We need to minimize the error here :

$$E_i = |y_i - (ax_i + b)|$$

$|y_i - (ax_i+b)| \leq E_i$ becomes $E_i = |y_i - (ax_i+b)|$ when bound becomes tight i.e., E_i is minimized.

$|y_i - (ax_i+b)| \leq E_i$:

- $y_i - (ax_i+b) \leq E_i$
- $y_i - (ax_i+b) \geq -E_i \Rightarrow -y_i + (ax_i+b) \leq E_i$

Formulation :

$$\text{Minimize } \sum_i E_i$$

Subject to :

$$y_i - (ax_i+b) \leq E_i \quad \forall\ i \in \{1,2,3,....n\}$$

$$-y_i + (ax_i+b) \leq E_i \quad \forall\ i \in \{1,2,3,....n\}$$

Note :

In general machine learning we define error as mean square error, hee we did it as absolute value error.

Other ways :

- We can also define as minimize $\max(E_i)$
- Cardinality of set of all points where $E_i = 0$

Integer Programming :

$$\text{Maximize } z = c^T \cdot x$$
$$\text{Subject to } A.x \le b \text{ ; } x \ge 0$$

$x \in \mathbb{R}$	Linear Programming
$x \in \mathbb{Z}$	Integer Programming
$x \in \{0,1\}$	Binary Integer Programming (BIP)
$x \in$ real and integer together	Multiple Integer Programming (MIP)

Integer Programming vs Linear Programming :

Let us do this considering an example :

$$\textbf{Maximize } 5x_1 + 8x_2$$

Subject to

$$x_1 + x_2 \le 6$$
$$5x_1 + 8x_2 \le 45$$
$$x_1, x_2 \ge 0 \text{ ; } x_1 \in \mathbb{R} \text{ ; } x_2 \in \mathbb{R}$$

Green part is the feasible region for the solution based on the constraints.

Point of intersection of both the lines maximizes the objective function from graphical interpretation. This point is (9/4,15/4) with a value of 41.25. This solution world for linear programming but not for integer programming.

If the constraint is now changed to $\mathbf{x_1 \in \mathbb{Z} \text{ ; } x_2 \in \mathbb{Z}}$. Let us look into why this in general is not trivial.

When $x_1 \in \mathbb{Z}$; $x_2 \in \mathbb{Z}$, there are many integer points in the feasible region, we have to find optimal sol point among all these. LP optima here is not an integer.

	LP*	Rounded off	Near Feasible point	IP*
x_1	2.25	2	2	0
x_2	3.75	4	3	5
Z	41.25	infeasible(does not satisfy 2nd constraint	$34 \neq z^*$	40

This proves that a small change in the constraint made the job difficult. IP optima was obtained through brute force. IP optimal solution that we obtained by rounding off LP optimal or making it feasible did not give us the right solution. Near FP is kind of LP relaxation. IP is difficult because of exhaustive search.

Note :

There may be situations where IP solution may not exist but LP solution does.

IP Formulation :

Example :

item	1	2	3	4	5	6	7	8	9	10
cost	200	300	250	500	700	100	250	350	800	600
val	5	6	8	2	9	7	5	2	1	5

Select a subset of the items within budget (1000) such that utility is maximum. If item 1 is selected then item 3 also should be selected. Either item 1 or item 3.

Let x_i is :

- 1 if ith item is selected
- 0 if ith item is not selected

We want to :

$$\text{Maximize } \sum_{i=1}^{10} v_i x_i$$

Constraint :

$$\sum_{i=1}^{10} c_i x_i \leq 1000$$

$$x_1 \leq x_3$$

$$x_1 + x_3 \leq 1$$

$$x_i \in \{0,1\}$$

Explanation :

- We want to maximize the utility, sum of all the values of the selected item should be maximized.
- Total cost of all the selected items should be less than 1000.
- If $x_1 = 1$ then $x_3 = 1$, if $x_1 = 1$ then $x_3 \in \{0,1\}$
- Only one among 1 and 3 are selected. Possible cases are :
  - If $x_1 = 1$ then $x_3 = 0$
  - If $x_3 = 0$ then $x_1 = 1$
- X_i can only have 2 values 0 and 1.

1. If item i is selected, then item j is also selected.	$x_i - x_j \leq 0$
2. Either item i is selected or item j is selected, but not both.	$x_i + x_j = 1$
3. Item i is selected or item j is selected or both.	$x_i + x_j \geq 1$
4. If item i is selected, then item j is not selected.	$x_i + x_j \leq 1$
5. If item i is not selected,then item j is not selected.	$x_j - x_i \leq 0$
6. At most one of items i, j, and k are selected.	$x_i + x_j + x_k \leq 1$
7. At most two of items i, j, and k are selected.	$x_i + x_j + x_k \leq 2$

8. Exactly one of items i, j and k are selected.	$x_i + x_j + x_k = 1$
9. At least one of items i, j and k are selected	$x_i + x_j + x_k \geq 1$

Formulate Graph Problems :

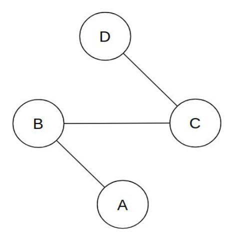
Consider a graph G = (V,E)

Maximum Independent set :

- Maximum size set such that no two vertices in it are connected by and edge.

Minimum Vertex Cover :

- Smallest set of vertices such that one end of every edge is a member of the set.

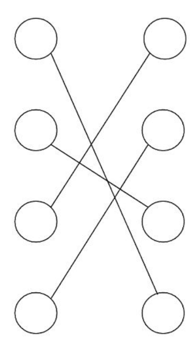
<table>
<tr><th>Maximum Independent Set</th><th>Minimum Vertex Cover</th></tr>
<tr><td colspan="2">$x_i = 0$ if vertex i is not selected
$= 1$ if vertex i is selected
Objective Function is number of vertices selected = $\sum_{v \in V} x_v$</td></tr>
<tr><td>$\text{Max} \sum_{v \in V} x_v$
Subject to :
$x_i + x_j \leq 1$ if $(i,j) \in E$
$x_i \in \{0,1\}$</td><td>$\text{Min} \sum_{v \in V} x_v$
Subject to :
$x_i + x_j \geq 1$ if $(i,j) \in E$
$x_i \in \{0,1\}$</td></tr>
<tr><td colspan="2"></td></tr>
<tr><td>2 => {A,C},{B,C},{B,D}</td><td>2 => {A,C},{A,D}{B,D}</td></tr>
</table>

Maximum Independent Set and Minimum Vertex Cover are vertex selection problems.

Maximum Bipartite Matching :

There are two sets of vertices (say jobs and people) We are interested in *Maximum Bipartite Matching* as a selection of edges that

- maximize the matching cost
- no two edges share the same vertex. (The same person does not get two jobs!)



$$\text{Max} \sum_{e \in E} w_e x_e$$

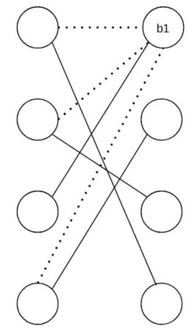
$$\sum_{e \in E,\, v \in e} w_e x_e \ ,\ v \in V$$

$$x_e \in \{0,1\}$$

Example :

We have two sets of vertices, one is jobs and the other is people. We have to find the best fits. Here out of all the 16 edges (4x4) we have to find the best 4 so that the assignment is the best. Same person does not get two jobs[no vertex with 2 edges].

Explanation :



$$x_e = 1 \text{ if edge e is selected}$$
$$= 0 \text{ if edge e is not selected}$$

We have to maximize the cost. So, the sum of weights of all the selected edges should be maximum.

Out of all the possible edges from a particular vertex lets say b1, only one edge should have $x_e = 1$ and rest others $x_e = 1$

Weight of all the dotted edges is 0 and the weight of the other one is 1.

Solving IP is hard :

Usually IP is hard to solve but not every IP problem is hard to solve.

Example : Find spanning tree of graph with minimum weight (polynomial time algo already exists for this). IP in general is NP hard but not every IP is hard.

When we encounter a new problem, we don't know if there already exists an efficient algorithm already, we can formulate in the form of IP/LP and solve (rapidly prototype an algorithm).

LP bound IP :

Consider the problem :

$$\text{Maximize } z = c^T \cdot x \text{ ; Subject to } A.x \le b$$

Let $(Z^*)_{LP}$ and $(Z^*)_{IP}$ are the optimal values of the objective function when solved with real and integer constraints. Then :

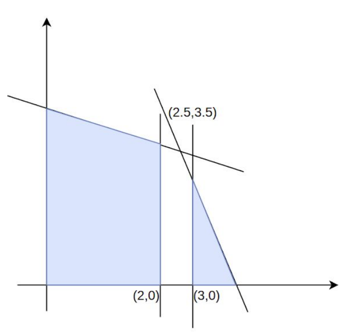
$$\mathbf{(Z^*)_{LP} \ge (Z^*)_{IP}}$$

Explanation :

We can never have an Integer point better than a real point. If so, then integer point itself becomes the optimal value in the real case as well. We cannot guarantee about $x^*_{LP} \ge x^*_{IP}$

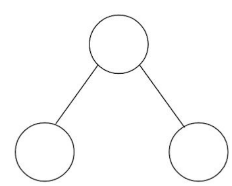
IP optima (max/min) is inferior to LP optima

Removing regions of feasible from further explore :



Here we are branching and rebounding the feasible region into another smaller one. Let us consider a case where the integer solution (3,3) is possible on the right side; this bounds us from further exploration on the right.

Instead of doing exhaustive search in the feasible region, we reduce the feasible region. From original problem we created two new problems (x<=2, x>=3)



Branch and Bound :

Let us take an example :

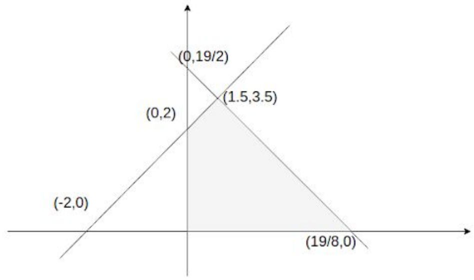
$$\mathbf{Maximize\ \ z, z = x_1 + x_2}$$

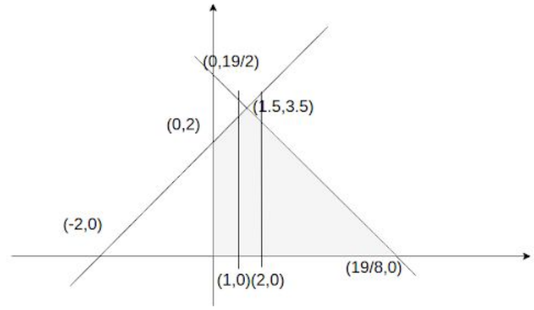
Subject to :

$$-x_1 + x_2 \le 2$$

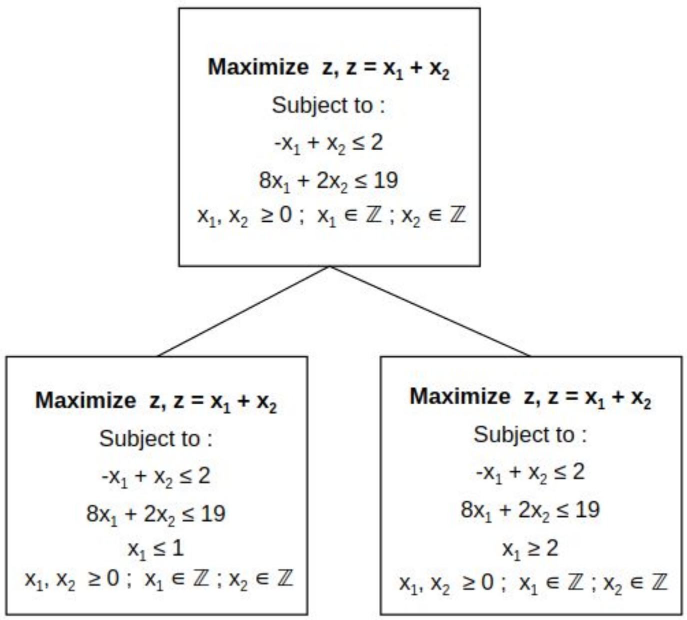
$$8x_1 + 2x_2 \leq 19$$

$$x_1, x_2 \geq 0 \; ; \; x_1 \in \mathbb{Z} \; ; x_2 \in \mathbb{Z}$$





We start just like how we are solving an LP problem. Optimal value is at (1.5,3.5). It cannot be the solution for IP. solution can only be integer so it can not be between x=1 and x=2. So we branched it into 2 parts(2 sub problems).



For x<=1 optimal solution is 4 at (1,3) $[Z_{LP} = Z_{IP}]$ and

For x>=2 optimal solution for LP is 3.5 at (2,1.5). This tells us that there can not be an integer point with optimal value better than 3.5.

Merging these two problems we get back to the actual problem ; optimal solution is the best one among optimal solutions of both the regions.

$$\Longrightarrow Z_{IP}^* = 4 \text{ at } (1,3)$$

When we create sub problems, we realize that some sub parts need not be further explored because there is already an integer solution available somewhere else. This helps in bounding and removing the need of branching that particular part until we get tiny sets.

In general, we start with the problem, we create 2 child problems and solve these two and check if we can end here. Or we further create sub problems of each sub problem with a known fact that IP optimal solution is always inferior to LP optimal solution. We keep branching until there is no feasible solution.

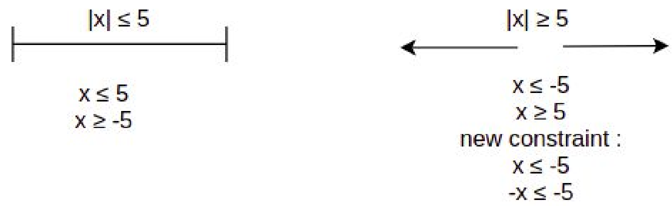
Example : https://www.ie.bilkent.edu.tr/~mustafap/courses/bb.pdf

On our first attempt itself if we land up in the integer solution then that is the optimal solution.

Adding integer variables to Constraints :

Example 1:

LP constraints such as |x|<= 5 is easy to incorporate because x lies between -5 and +5.



To incorporate new constraint in LP formulations :

- By introducing integer variables.

$$x \leq 5$$

$$-x \leq -5$$

We make $x \leq 5$ this trivially true by adding M (a very large quantity) . Now, $x \leq 5+M$ is always true. Similarly to $-x \leq 5$ and make it $-x \leq 5+M$. Now, both are trivially true. We want one of them to be active at a time. We introduce an integer variable y.

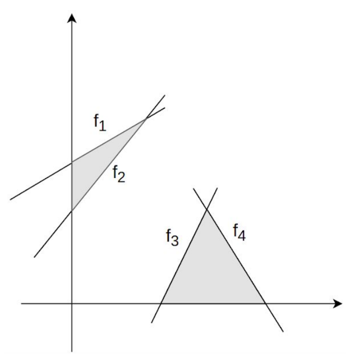
$$x \leq 5+y.M$$

$$-x \leq -5+(1-y).M$$

$$y \in \{0,1\}$$

When y=0 => y.M = 0, and equation $x \leq 5+y.M$ is fully active and $-x \leq -5+(1-y).M$ becomes $-x \leq -5+M$ and $-x \leq -5+(1-y).M$ becomes trivially true. For any value of y only one of them will be active at a time.

Example 2:



We want either the region where f_1 and f_2 are active or the region where f_3 and f_4 are active.

Let constraints be :

- $f_1(x_1,x_2) \leq b_1$
- $f_2(x_1,x_2) \leq b_2$
- $f_3(x_1,x_2) \leq b_3$
- $f_4(x_1,x_2) \leq b_4$

We introduce a variable y :

- $f_1(x_1,x_2) \leq b_1 + y.M$
- $f_2(x_1,x_2) \leq b_2 + y.M$
- $f_3(x_1,x_2) \leq b_3 + (1-y).M$
- $f_4(x_1,x_2) \leq b_4 + (1-y).M$
- $y \in \{0,1\}$

We can also introduce multiple integer variables and put an additional constraint on y based on our requirement.

Example 3 :

Rewrite the problem as IP

$$\textbf{Maximize } Z = c_1x_1 + c_2x_2$$

Subject to strictly two of the three are true :

$$a_1x_1 + a_2x_2 \le b_1 + y_1M$$
$$a_3x_1 + a_4x_2 \le b_2 + y_2M$$
$$a_5x_1 + a_6x_2 \le b_3 + y_3M$$

And

$$|x_1| \ge b_4$$
$$|x_2| \le b_5$$

Solution :

$$\text{Maximize } Z = c_1x_1 + c_2x_2$$

Subject to :

$$a_1x_1 + a_2x_2 \le b_1 + y_1M$$
$$a_3x_1 + a_4x_2 \le b_2 + y_2M$$
$$a_5x_1 + a_6x_2 \le b_3 + y_3M$$

$$x_1 \le -b_4 + y_4M$$
$$-x_1 \le -b_4 + (1 - y_4)\, M$$
$$x_2 \le b_5$$
$$-x_2 \le b_5$$

$$y_1, y_2, y_3 \in \{ 0,1 \}$$
$$y_1 + y_2 + y_3 = 1$$
$$M >> 0, \text{ M is a large positive integer v}$$
$$y_4 \in \{ 0,1 \}$$
$$x_1, x_2 \in \mathbb{Z}$$

LP Relaxation :

A popular way of working with IP problems is with **LP Relaxation**

- **Core Idea :** Relax the integral constraint and create a linear programming problem (LP).
- *Round* and get an integral solution.
  - Be happy with the integral solution (after rounding off), even if it need not be optimal.
  - Be happy with the sub-optimal solution. But have some theoretical "bounds" on how bad/worse is this.
  - Get integral solutions that are optimal itself (lucky cases!)
- Often is of interest, not for the solution, but for the associated theoretical results.

"This relaxation technique transforms an NP-hard optimization problem (integer programming) into a related problem that is solvable in polynomial time (linear programming); the solution to the relaxed linear program can be used to gain information about the solution to the original integer program."

Reasons :

LP might give integer solutions, rounding off and finding nearest integer solutions.

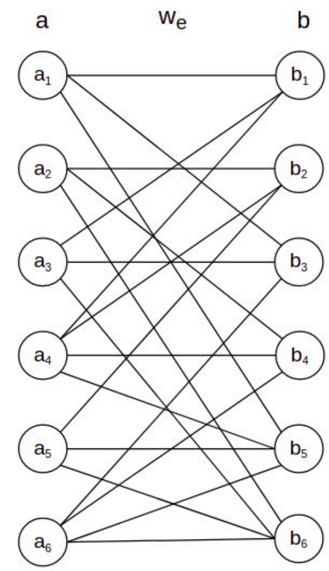
Example :

$$x_e \in \{0,1\} \text{ ----LPR---> } 0 \le x_e \le 1$$

Formulation using LP Relaxation :

Maximum Bipartite Matching :

Selection of edges (x_e is a selection variable) such that the sum of weights of all selected edges is maximized and every vertex at least one edge is selected.



$$\text{Max} \sum_{e \in E} w_e x_e$$

$$\sum_{e \in E,\, v \in e} w_e x_e \ , \ v \in V$$

$$x_e \in \{0,1\}$$

$$x_e \in \{0,1\} \xrightarrow{\text{LPR}} 0 \leq x_e \leq 1$$

Let us look into how good/bad is LP solution w.r.t. IP optima and how to get integral optima for LP optima.

On LP Relaxation :

Now, x_e is somewhere in between 0 and 1. Rounding off may not satisfy the constraints.

Consider the vertex a_1 one among the three edges is not not 0 or 1 :

- At any vertex if on of the edge is not saturated i.e., $x_e \notin \{0,1\}$, then there should be atleast one more of the same type.
    - If $(a_1,b_1) = 0.5$ then one of them should be 0.5 or both of them together should be 0.5.
    - Let us assume that (a_1,b_3) is not saturated then, (a_3,b_3) or (a_6,b_3) or both are not saturated. Therefore, (a_3,b_1) also maynot be saturated and same with (a_1,b_1).
    - Looking at the sequence of edges, that start from a vertex in a say a_1 and goes to b_2 and comes back to a_3, then to b_4 again and finally to a_1.

$$a_1\ b_2\ a_3\ b_4\ a_1$$

$$a_1\ b_2\ a_3\ b_4\ a_1$$

$$e_1\ e_2\ e_3\ e_4\ \ldots\ldots\ e_t$$

    - e_t is always even.
- If the solution is not integral then there exists at least 1 cycle and the cycle has an even number of edges.
- We modify edges as
    - $y_e = x_e - \varepsilon$; if e is odd
    - $y_e = x_e + \varepsilon$; if e is even
- Net weight of edges is the same.
- Now, cost(y) :

$$\text{cost}(y) = \text{cost}(x) + \varepsilon \sum_{i=1}^{t} (-1)^i w_{ei}$$

$$\Delta = \sum_{i=1}^{t} (-1)^i w_{ei}$$

$$\text{cost}(y) = \text{cost}(x) + \varepsilon.\Delta$$

- If $\Delta > 0$, then cost(y) > cost(x). But the x that we obtained is through LP and there can not be a y better than x. Because cost(x) is the best we obtained and cost(y) $\ngtr$ cost(x) for a maximization problem. If $\Delta < 0$, if we pick an $\varepsilon < 0$, but the cost(y) again could have gone up. Now, the only possibility left out for ε is $\Delta = 0$.

$$\text{cost}(y) = \text{cost}(x)$$

- Therefore, if we find such a cycle and if we increase and decrease ε for odd and even edges then values of selection variables x_e cost will not change. We can always find such a cycle and find an ε such that at least one edge will become 0 or 1. A non integral edge will become an integral edge and the cycles break. We repeat doing this, we get an integral solution with the same objective.

Conclusion :

LP Relaxation helped us to solve and get the solution efficiently. This is a smart way to round without changing the objective.

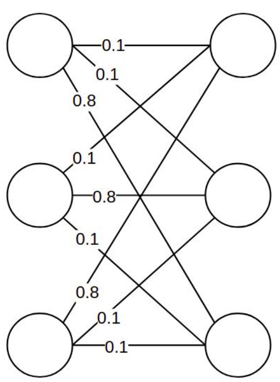
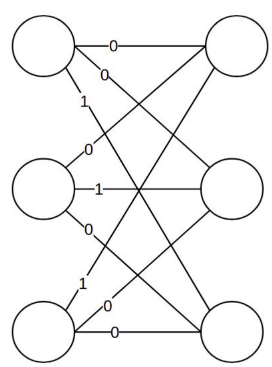
Steps :

- Formulate on IP
- LP relaxation
- Solve LP and get x*
- X is not integer. We can round x to integer with no reduction in z.

Note :

We only increase ε until when one edge becomes 1 and minus edge becomes 0. At Least one of them will become 0 or 1.

Example :

	
$\sum_{e \in E} w_e x_e = 0.8*10*3 + 0.1*10*6 = 30$	$\sum_{e \in E} w_e x_e = 1*10*3 + 0*10*6 = 30$

Minimum Vertex Cover :

$$\text{Min} \sum_{v \in V} y_v$$

Subject to :

$$y_u + y_v \geq 1 \text{ if } (u,v) \in E$$
$$y_u \in \{0,1\}$$

Objective function is to minimize cardinality of the set of vertices selected. Min $\sum_{v \in V} y_v = |S_{OPT}|$

On LP Relaxation :

$$\text{Min} \sum_{v\in V} x_v$$

Subject to :

$$x_u + x_v \geq 1 \text{ if } (u,v) \in E$$
$$x_u \in [0,1]$$

Let the solution be x_u. Min $\sum_{v\in V} x_v = Z_{LP}*$

Vertex cover :

$$S_{LP} = \{ v \mid x_v* \geq ½ \}$$

$x_u + x_v \geq 1$ then at least one of the should be ½ .

We know that $|S_{OPT}| \leq |S_{LP}|$.

Since this is a minimization problem we know that $|Z_{LP}*| \leq |Z_{IP}*|$

$$|S_{LP}| = (\sum_{v\in S_{LP}} 1) \leq (2* \sum_{v\in S_{LP}} x_v*) \leq (2* \sum_{v\in V} x_v*) \leq (2* \sum_{v\in V} y_v*) = 2*|S_{OPT}|$$

$$2*|Z_{IP}*| \leq 2*|Z_{LP}*|$$

$$|S_{OPT}| \leq |S_{LP}| \leq 2*|S_{OPT}|$$

The vertex cover that we obtained on LP Relaxation is bound by optimal vertex cover by IP and twice the optimal vertex cover by IP. we have a bound on the solution we obtained through LP Relaxation. In some problems when we do LPR and solve we can get some bound or measure of how superior the solution w.r.t optima.

The results are not good but there is bound on how bad the solution is.

Formulation : Fitting line

Norms :

Link : https://rorasa.wordpress.com/2012/05/13/l0-norm-l1-norm-l2-norm-l-infinity-norm/

Note:

- $||e||_p = (|e_1|^p+|e_2|^p+\ldots+|e_n|^p)^{1/p}$
- $||e||_\infty = \max(|e_1|,|e_2|,\ldots,|e_n|)$
- $||e||_0 = \#\text{nonZero}(|e1|,|e2|,\ldots,|en|)$

We are given a set of N points (xi,yi), and we are asked to fit (or find) a line (say $ax+b=y$) that minimize an "error" (in predicting). i.e., find a and b by minimizing an error vector

$$e^T = [e_1\ e_2 \ldots\ldots e_n]$$
$$ei = yi - (ax_i+b)$$

We want to know which optimizes what. Notion of objective function is important.

We want to minimize e. We also want to know about, in which form it (norm) has to be minimized.

L1 Norm :

Following the definition of norm, l_1-norm of x is defined as $\|x\|_1 = \Sigma_i |x_i|$

L1 Optimization :

This is LP formulation.

We need to minimize the error here (L1 Norm):

$$e_i = |y_i - (ax_i+b)|$$

Here we are minimizing distance along y.

$|y_i - (ax_i+b)| \le E_i$ becomes $E_i = |y_i - (ax_i+b)|$ when bound becomes tight i.e., e_i is minimized.

$|y_i - (ax_i+b)| \le E_i$:

- $y_i - (ax_i+b) \le E_i$
- $y_i - (ax_i+b) \ge -E_i \implies - y_i + (ax_i+b) \le E_i$

Formulation :

$$\text{Minimize } \Sigma_i e_i$$

Subject to :

$$y_i - (ax_i+b) \le e_i \quad \forall\, i \in \{1,2,3,....n\}$$
$$- y_i + (ax_i+b) \le e_i \quad \forall\, i \in \{1,2,3,....n\}$$

L2 Norm :

Following the definition of norm, l_2-norm of x is defined as $\|x\|_2 = (\Sigma_i x_i^2)^{1/2}$

L2 Optimization :

We need to minimize the error here (L2 Norm):

$$e_i = y_i - (ax_i+b)$$

Error is *mean square error*.

Formulation :

$$\text{Minimize } \|x\|_2$$

Subject to :

$$Ax = b$$

This gives us a closed form solution. We will look into this later.

L∞ Norm :

Maximum error wrt any of the points. This is LP formulation.

As always, the definition for L∞-norm is

$$||x||_\infty = \sqrt[\infty]{x_j^\infty}$$

Now this definition looks tricky again, but actually it is quite straightforward. Consider the vector x, let's say if x_j is the highest entry in the vector x, by the property of the infinity itself, we can say that

$$x_j^\infty \gg x_i^\infty \quad \forall\, i \neq j$$
$$\text{then } \sum_i x_i^\infty = x_j^\infty$$
$$||x||_\infty = \sqrt[\infty]{\sum_i x_i^\infty} = \sqrt[\infty]{x_j^\infty} = |x_j|$$
$$||x||_\infty = \max(|x_j|)$$

that is the maximum magnitude among all the entries of the vector.

$$\|e\|_\infty = \max(|e_j|)$$

L∞ Optimization :

Here we want to minimize the maximum error. We want everybody to be as close as possible.

Formulation :

$$\text{Min } t \,|_{a,b}$$

Subject to :

$$e_i \leq t \quad \forall\, i \in \{1,2,3,....n\}$$

$$-e_i \leq t \quad \forall\, i \in \{1,2,3,....n\}$$

Standard form :

L0 Norm :

Noof non zero elements

$$||x||_0 = \#(i \,|\, x_i \neq 0)$$

L0 Norm is sparse.

L0 Optimization :

The sparsest solution means the solution which has fewest non-zero entries, i.e. the lowest L0-norm. A line that passes through most number of points. This problem can be formulated as an integer programming (IP) and is highly non-convex. L0-minimisation is regarded as an NP-hard problem.

- A popular strategy is to simplify this as a L1. We relax L0 to L1
- Under certain constraints, situations, L1 norm optimization will give solutions that are also sparse like L0.

Note :

In many cases, the l_0-minimisation problem is relaxed to be a higher-order norm problem such as l_1-minimisation and l_2-minimisation.

Example :

Minimize

$$||Ax - b||_1$$

Subject to :

$$||x||_\infty \leq 1$$

Re writing the above problem as :

$$\min\left(\sum_{i=1}^{n} y_i\right)$$

Subject to :

$$-y_i \leq \sum_{j=1}^{n}((a_i x_i) - b_i) \leq y_i \text{ , } i = 1, 2, 3, \ldots, m$$

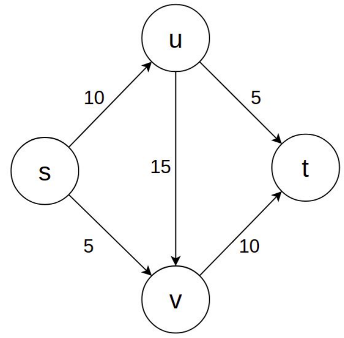
$$-1 \leq x_j \leq 1 \text{ ; } j = 1, 2, 3, \ldots, n$$

Max Flow Problem :

We have a source and a target. We want maximum flow possible from source (S) to target (T). LP Problem

Constraints are :

- Capacity
  - Constant ; can not be more than edge capacity ; maximum possible capacity is edge capacity
  - Non negative.
- No storage
  - Net flow is constant



Formulation :

Max Flow Problem

$$\text{Max } f_{s,u} + f_{s,v}$$

subject to

$$f_{s,u} = f_{u,v} + f_{u,t}$$
$$f_{s,v} + f_{u,v} = f_{v,t}$$
$$0 \leq f_{s,u} \leq 10$$
$$0 \leq f_{s,v} \leq 5$$
$$0 \leq f_{u,t} \leq 5$$
$$0 \leq f_{u,v} \leq 15$$
$$0 \leq f_{v,t} \leq 10$$

$f_{s,u} + f_{s,v}$ = amount of outflow from source

We want to maximize the amount of outflow from source or we can also write this as max inflow to target.

Net flow at a vertex is constant :

- $f_{s,u} = f_{u,v} + f_{u,t}$
- $f_{s,v} + f_{u,v} = f_{v,t}$

Standard Formulation :

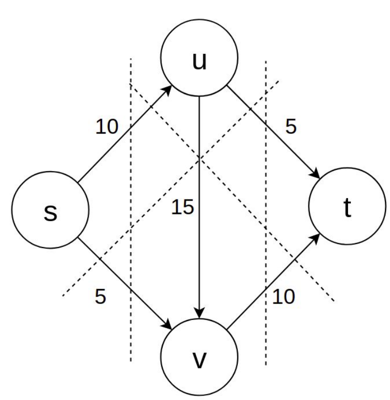
We can write this problem as :

$$\text{Max } c^T \cdot x : \text{Subject to } A.x \leq b, x \geq 0$$

Min-Cut Problem :

We want to cut this entire graph into 2 parts such that

- a set of vertices are with (S) and rest with (T).
- No flow from (S) to (T)



This is an IP Problem.

Max Flow and Min Cut are Equivalent

But one seems to be IP and the other seems to be IP, but they are equal. So, the complexity can not be very different. These are dual problems, problems come in pairs.

Formulation :

$$Min\ 10y_{su} + 5y_{sv} + 15y_{uv} + 5y_{ut} + 10y_{vt}$$

Subject to:

$$y_{su} + u_u \geq 1$$
$$y_{sv} + u_v \geq 1$$
$$y_{uv} - u_u + u_v \geq 0$$
$$y_{ut} - u_u \geq 0$$
$$y_{vt} - u_v \geq 0$$
$$y_i \in \{0,1\}$$
$$u_i \in \{0,1\}$$

Explanation :

- For every edge there is a corresponding y variable.
- y_i is 1 means it is cut and $y_i = 0$ means it is not cut.
- u_u is 1 if u is in the cut with set S else u_u is 0.
- u_v is 1 if v is in the cut with set S else u_v is 0.
- $y_{su} + u_u \geq 1 \Longrightarrow$ at least one of them should be equal to 1. Both being 0 is not acceptable.
  - su is cut and u is on the same side of S.
  - su is cut and u is not on the same side of S.
  - If y_{su} is not cut and u is on the same side of S.
- $y_{uv} - u_u + u_v \geq 1 \Longrightarrow$ when you cut the edge connecting u and v they have to be on different sides or if u is in S and v is not then uv should be cut.

Duality :

Example :

Let us consider an LP an problem :

$$\textbf{Maximize } 2x_1 + 3x_2$$

Subject to

1. $4x_1 + 8x_2 \leq 12$
2. $2x_1 + x_2 \leq 3$
3. $3x_1 + 2x_2 \leq 4$
4. $x_1, x_2 \geq 0$; $x_1 \in \mathbb{R}$; $x_2 \in \mathbb{R}$

We are interested in solving these problems without actually solving completely.

Procedure :

On combining equation 1 and objective function we get :

$$2x_1 + 3x_2 \leq 4x_1 + 8x_2 \leq 12$$

This a quite weaker relation. We can write the equation with a tighter bound :

$$2x_1 + 3x_2 \leq ½*(4x_1 + 8x_2) \leq 6.$$

Make it better by adding equation 1 and equation 2 :

$$2x_1 +3\, x_2 + 4x_1 + 8x_2 \leq 15$$

$$3*(2x_1 + 3x_2) \leq 15$$

$$2x_1 + 3x_2 \leq 5$$

$$\Longrightarrow 2x_1 + 3x_2 \leq 2x_1 + 3x_2 \leq 5$$

Here, we made the bound much tighter and solved the problem without actually solving

Let us say we have dual variables. y_1, y_2 ,y_3

1. $4x_1 + 8x_2 \leq 12(y_1)$
2. $2x_1 + x_2 \leq 3 \ (y_2)$
3. $3x_1 + 2x_2 \leq 4 \ (y_3)$

We multiply constraint i by y_i . if they are non negative variables requirity sign does not change such that the bounds of all the constraints become as tight as possible.

Writing this problem in terms of y_i's :

- Multiply ith constraint by y_i.
- If $y_i \geq 0$ then the inequality sign does not change.
- We want to add the results such that bounds become as tight as possible

$$\Longrightarrow \text{minimize } y = 12y_1 + 3y_2 + 4y_3$$

Subject to :

$$4y_1 + 2y_2 + 3y_3 \geq 2$$

$$8y_1 + y_2 + 2y_3 \geq 3$$

$$y_1, y_2, y_3 \geq 0$$

Explanation :

This is a maximization problem, bound has to be tight so y is a minimization problem [we are looking for such a smallest y].

$$12y_1 + 3y_2 + 4y_3 \geq y_1\,(4x_1 + 8x_2) + y_2(\,2x_1 + x_2) + y_3\,(3x_1 + 2x_2)$$

$y_1\,(4x_1 + 8x_2) + y_2(\,2x_1 + x_2) + y_3\,(3x_1 + 2x_2)$ has to bound $2x_1 + 3x_2$

$$\Longrightarrow 4y_1 + 2y_2 + 3y_3 \geq 2$$

$$\Longrightarrow 8y_1 + y_2 + 2y_3 \geq 3$$

$$y_1, y_2, y_3 \geq 0$$

We created a new LP problem from one LP problem. Problem in x is primal and problem in y is dual. This is the dual nature of LP.

Generalization :

For every primal LP problem in the form of

$$\text{Maximize } c \cdot x$$
$$\text{Subject to } A.x \leq b,\ x \geq 0,$$

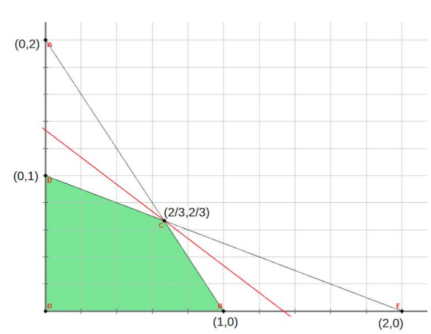
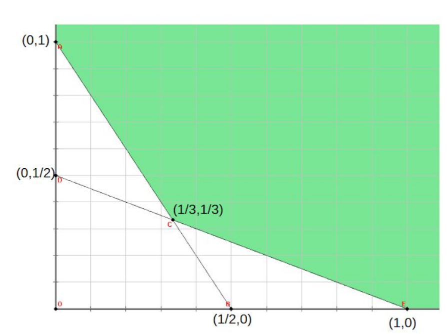
there exists an equivalent dual LP problem

$$\text{Minimize } b \cdot y$$
$$\text{Subject to } A^{T}.y \geq c,\ y \geq 0.$$

This property of LP can be used to show many important theorems. For instance, the max-flow min-cut theorem can be proven by formulating the max-flow problem as the primal LP problem.

$$\begin{array}{ccc} \text{Primal} & \Leftrightarrow & \text{Dual} \\ \text{Maximize } c \cdot x & \Leftrightarrow & \text{Minimize } b \cdot y \\ \text{Subject to } A.x \leq b,\ x \geq 0 & \Leftrightarrow & \text{Subject to } A^{T}.y \geq c,\ y \geq 0 \end{array}$$

Example :

Primal : $\text{Maximize } x_1 + x_2$ Subject to : $2x_1 + x_2 \leq 2$ $x_1 + 2x_2 \leq 2$ $x_1, x_2 \geq 0;\ x_1 \in \mathbb{R};\ x_2 \in \mathbb{R}$	Dual : $\text{Minimize } 2y_1 + 2y_2$ Subject to : $2y_1 + y_2 \geq 1$ $y_1 + 2y_2 \geq 1$ $y_1, y_2 \geq 0;\ y_1 \in \mathbb{R};\ y_2 \in \mathbb{R}$
	
z * is at (⅔,⅔) with a value of 4/3	z * is at (⅓ ,⅓) with a value of 4/3

Primal to Dual :

Let us look unto these solutions, dual is tight to primal or is it the other way round? We say that a constraint is tight when inequality becomes equality.

Dual and Primal have the same z* for LP, not for IP.

No of variables in primal becomes no of constraints in dual and vice versa.

Writing a primal when a :

Primal	Dual
$x_1, x_2, x_3, \ldots\ldots x_n$	$y_1, y_2, y_3, \ldots\ldots y_m$
A	A^T
b	c
c	b
Max $c^T X$	Min $b^T Y$
$\leq$	$y_i \geq 0$
$\geq$	$y_i \leq 0$
$=$	$y_i \in \mathbb{R}$
$x_j \geq 0$	j^{th} constraint $\geq$
$x_j \leq 0$	j^{th} constraint $\leq$
$x_j \in \mathbb{R}$	j^{th} constraint $=$

Theoretical results :

Primal :	Dual :
Maximize $c^T \cdot x$ Subject to $A.x \leq b$ $x \geq 0$	Minimize $b^T \cdot y$ Subject to $A^T y \geq c$ $y \geq 0$

Let us look at the relation between feasible points for both the problems. Let x, y be feasible points in P and D.

$$c^T x = x^T c \leq x^T A^T y = (Ax)^T y \leq b^T y$$

$$\Longrightarrow c^T x \leq b^T y$$

This relation holds for all the feasible points.

$c^T x \leq b^T y$ --- *Weak Duality*

Strong Duality (LP) ----- $c^T x^* = b^T y^*$. This is true for LP.

Let us assume that we solved this as LP and IP :

In maximization problem

$$z^*_{IP} \leq z^*_{LP} \Longrightarrow c^T x^*_{IP} \leq c^T x^*_{LP}$$

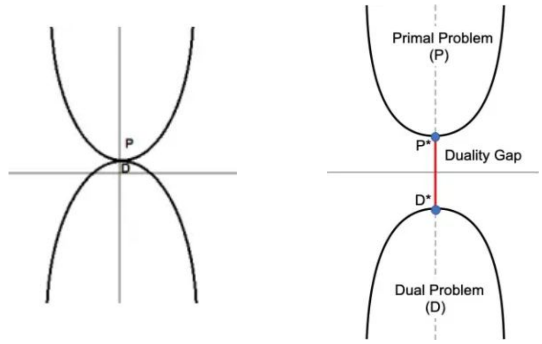
$$c^T x^*_{IP} \leq c^T x^*_{LP} = b^T y^*_{LP}$$

$$c^T x^*_{IP} \leq c^T x^*_{LP} = b^T y^*_{LP} \leq b^T y^*_{IP}$$

$$c^T x^*_{IP} \leq b^T y^*_{IP}$$

- IP only has Weak Duality.
- LP has Strong Duality at Optimal Value and Weak Duality holds for any feasible solution.
- IP has Duality gap

For LP vs IP Duality gap visualisation :



Primal and dual have no duality gap when primal and dual have the same optimal solution. There may be many optimal solutions. Our intention here is to know that there are

- Two different views/perspectives of a single problem.
- This helps us design class of Primal-Dual Algos
- Theoretical results (weak/strong Duality) is important

Example : max flow - Min Cut.

LP duality Results :

Duality Theorem for Linear Programming :

For the linear programs

$$\text{Maximize } c^T \cdot x : \text{Subject to } A.x \leq b, x \geq 0 \qquad (P)$$

and

$$\text{Minimize } b^T \cdot y : \text{Subject to } A^T y \geq c,\ y \geq 0 \qquad (D)$$

exactly one of the following possibilities occurs:

1. Neither (P) nor (D) has a feasible solution.
2. (P) is unbounded and (D) has no feasible solution.
3. (P) has no feasible solution and (D) is unbounded.
4. Both (P) and (D) have a feasible solution. Then both have an optimal solution, and if x* is an optimal solution of (P) and y' is an optimal solution of (D), then $c^T x^* = b^T y^*$
5. *Maximum* (P) = *minimum* (D)

Primal feasible : Maximize $z = 2x_1 + x_2$ Subject to : $x_1 + x_2 \leq 4$ $x_1 - x_2 \leq 2$ $x_1, x_2 \geq 0$	*Dual feasible :* Minimize $4y_1 + 2y_2$ Subject to : $y_1 + y_2 \geq 2$ $y_1 - y_2 \geq 1$ $y_1, y_2 \geq 0$
Primal feasible and unbounded : Maximize $z = 2x_1 + x_2$ Subject to : $x_1 - x_2 \leq 4$ $x_1 - x_2 \leq 2$ $x_1, x_2 \geq 0$	*Dual infeasible :* Minimize $4y_1 + 2y_2$ Subject to : $y_1 + y_2 \geq 2$ $-y_1 - y_2 \geq 1$ $y_1, y_2 \geq 0$

Primal infeasible :	*Dual feasible and unbounded :*
Maximize $z = 2x_1 + x_2$ Subject to : $-x_1 - x_2 \leq -4$ $x_1 - x_2 \leq 2$ $x_1, x_2 \geq 0$	Minimize $-4y_1 + 2y_2$ Subject to : $-y_1 + y_2 \geq 2$ $-y_1 - y_2 \geq 1$ $y_1, y_2 \geq 0$
Primal infeasible :	*Dual infeasible :*
Maximize $z = 2x_1 + x_2$ Subject to : $-x_1 + x_2 \leq -4$ $x_1 - x_2 \leq 2$ $x_1, x_2 \geq 0$	Minimize $-4y_1 + 2y_2$ Subject to : $-y_1 + y_2 \geq 2$ $y_1 - y_2 \geq 1$ $y_1, y_2 \geq 0$

Complementary Slackness :

Primal :	Dual :
Maximize $c^T \cdot x$ Subject to $A.x \leq b$ $x \geq 0$	Minimize $b^T \cdot y$ Subject to $A^T y \geq c$ $y \geq 0$

For any two feasible points of x and y :

Weak Duality says that primal objective will not be better than dual objective.

$$c^Tx \leq b^Ty$$

Strong Duality says that at the optima, the objective of primal and dual are the same.

$$c^Tx^* = b^Ty^*$$

Result :

$$c^Tx^* \leq (A^T y)^T x^* = y^TAx^* \leq y^{*T}b$$

Result a :

From strong duality we know that $c^Tx^* = y^{*T}b$

Therefore, $c^Tx^* = (A^T y)^T x^*$

Transpose on both sides :

$$x^{*T}c = x^{*T}A^T y$$

$$x^{*T}[c - A^T y] = 0 \qquad (1)$$

Result b :

$$y^TAx^* = y^{*T}b$$

$$y^T[Ax^* - b] = 0 \qquad (2)$$

From equation (1) :

$$x^{*T} = 0 \text{ or } [c - A^T y] = 0$$

i.e., $x^* = 0$ (variable is 0 in primal) or the constraint is tight in dual. There is no slack on both sides. Similarly for equation 2. Either Dual variable is 0 or the corresponding constraint in primal is tight.

Example :

$$\text{Max } c^T.x$$

$$Ax \leq b$$

At optimal points : $c^T.x^* = b^T.y^*$

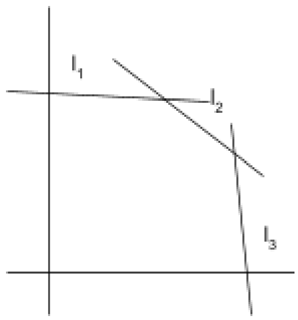
Assume we increase(relax b by) b by ε. When the constraint is not tight we change(relax) constraint by a very small amount and this will not have any effect on $c^T.x^* = b^T.y^*$. This is only possible when $y^*=0$.

Example :

$$\text{Max } c^T.x$$

Subject to :

$$l_1(x) \le b_1$$
$$l_2(x) \le b_2$$
$$l_3(x) \le b_3$$



Let the point of intersection between l1 and l2 be optima. Let the corresponding dual variables be y_1, y_2, and y_3. y_3 is 0. Because either the constraint is tight or the dual variable corresponding to the constraint is 0.

Example - Diet vs Pills :

Primal : There are *n* foods, *m* nutrients, and a person (the buyer) is required to consume at least b_i units of nutrient *i* (for $1 \le i \le m$) *. Let* $A = [a_{ij}]$ denote the amount of nutrient *i* present in one unit of food *j*. Let c_i denote the cost of one unit of food item *i*. One needs to design a diet of minimal cost that supplies at least the required amount of nutrients.	Dual : Consider a seller supplying the nutrients directly through pills (eg. vitamin pills). The seller wants to charge as much as he can for the pills, but the buyer will have to come to him for pills. i.e., constraint: the price of pills is such that it is never cheaper to buy a food in order to get the nutrients in it rather than buy the nutrients directly. If y_i is the vector of nutrient prices, this gives the constraints $A^T y \le c$.
$\text{Min } \sum_{i=1}^{n} c_i x_i$ Subject to : $x_i \sum_{j=1}^{m} a_{ij} \ge b_i$ $x \ge 0$	$\text{Max } \sum_{i=1}^{m} b_i y_i$ Subject to : $y_i \sum_{j=1}^{n} a_{ij} \ge c_i$
Explanation : $x_i = 0$ if ith food is selected $= 1$ if ith food is not selected Total cost of selected foods should be minimized. $\sum_{j=1}^{m} a_{ij} \ge b_i$ is $Ax \ge b$	Explanation : $y_i = 0$ if ith food is selected $= 1$ if ith food is not selected Total cost of selected nutrients should be maximized. $\sum_{j=1}^{n} a_{ij} \le c_i$ is $A^T y \le c$

Example - Student-Shop-Supplier :

A health conscious student wants to buy (fractional portions of) Brownie and CheeseCake from a shop to meet the "Requirements" at the lowest cost.

	Chocolate	Sugar	Cream Cheese	Cost
Brownie	3	2	2	50
CheeseCake	0	4	5	80
Requirements	6	10	8	

A supplier wants to supply the raw items to the shop so as to maximize the income.

A student is deciding what to purchase :
brownies, which cost Rs 50 each, and cheesecakes, which cost Rs 80. Shop is fine with purchase fraction items. Both items use chocolate, sugar and cream cheese in different proportions.
Being health conscious, the student has decided that she needs at least six total units of sugar, along with ten units of sugar and eight units of cream cheese. She wishes to optimize her purchase by finding the least expensive combination of brownies and cheese cakes that meet these requirements.

Consider the perspective of the Supplier who supplies the shop with the chocolate, sugar, and cream cheese needed to make the snacks. Shop tells the supplier that it needs at least 6 units of chocolate, 10 units of sugar, and 8 ounces of cream cheese, to meet the students minimum nutritional requirements.
Table is shared. The supplier now solves the following optimization problem : How can I set the prices per unit of chocolate, sugar, and cream cheese so that the baker will buy from me, and so that I will maximize my income?

$$\text{Min } 50x_1 + 80x_2$$

Subject to :

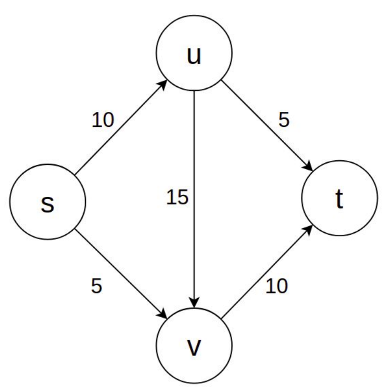
$$3x_1 \geq 6$$
$$2x_1 + 4x_2 \geq 10$$
$$2x_1 + 5x_2 \geq 8$$

$$\text{Max } 6y_1 + 10y_2 + 8y_3$$

Subject to :

$$3y_1 + 2y_2 + 2y_3 \geq 50$$
$$0y_1 + 4y_2 + 5y_3 \geq 80$$

Example - Max Flow vs Min Cut :



$$\text{Max } f_{s,u} + f_{s,v}$$

$$Min\ 10y_{su} + 5y_{sv} + 15y_{uv} + 5y_{ut} + 10y_{vt}$$

<table>
<tr>
<td>subject to
$$f_{s,u} = f_{u,v} + f_{u,t}$$
$$f_{s,v} + f_{u,v} = f_{v,t}$$
$$0 \leq f_{s,u} \leq 10$$
$$0 \leq f_{s,v} \leq 5$$
$$0 \leq f_{u,t} \leq 5$$
$$0 \leq f_{u,v} \leq 15$$
$$0 \leq f_{v,t} \leq 10$$</td>
<td>Subject to:
$$y_{su} + u_u \geq 1$$
$$y_{sv} + u_v \geq 1$$
$$y_{uv} - u_u + u_v \geq 0$$
$$y_{ut} - u_u \geq 0$$
$$y_{vt} - u_v \geq 0$$
$$y_i \in \{0,1\}$$
$$u_i \in \{0,1\}$$</td>
</tr>
<tr>
<td>$$\text{Max } c^T.x$$
Subject to :
$$Ax \leq b$$</td>
<td>$$\text{Min } b^T y$$
Subject to :
$$A^T y \geq c$$</td>
</tr>
<tr>
<td colspan="2">$$\begin{array}{c|ccccc|c|c} b & 1 & 1 & 0 & 0 & 0 & & c \\ \hline A & x_1 & x_2 & x_3 & x_4 & x_5 & & \\ \hline y_1 & 1 & 0 & 0 & 0 & 0 & & 10 \\ y_2 & 0 & 1 & 0 & 0 & 0 & & 5 \\ y_3 & 0 & 0 & 1 & 0 & 0 & \leq & 15 \\ y_4 & 0 & 0 & 0 & 1 & 0 & = & 5 \\ y_5 & 0 & 0 & 0 & 0 & 1 & \geq & 10 \\ y_6 & 1 & 0 & -1 & -1 & 0 & & 0 \\ y_7 & 0 & 1 & 1 & 0 & -1 & & 0 \end{array}$$</td>
</tr>
</table>

<u>Example - Max Matching vs Min Vertex Cover :</u>

<table>
<tr>
<td>A matching is a sub graph in which no two edges share a common node. Maximum matching problem asks for a maxi-mum matching given any graph.
Maximize the sum of weights of edges selected
$$\text{Max } I^T x$$
$$\begin{bmatrix} & \text{incident matrix} & \end{bmatrix}_{\text{vxe}} \begin{bmatrix} \\ \\ \\ \end{bmatrix} \leq \begin{bmatrix} 1 \end{bmatrix}$$
This says that for every vertex there to be an edge that is selected</td>
<td>A vertex cover of a graph is a set of vertices that includes at least one endpoint of every edge of the graph. A minimum vertex cover is a vertex cover having the smallest possible number of vertices for a given graph. Minimize size of vertex set selected.
$$\text{Min } I^T y$$
$$\begin{bmatrix} \\ \\ \\ \end{bmatrix}_{\text{exv}} \begin{bmatrix} y \end{bmatrix} \geq \begin{bmatrix} 1 \end{bmatrix}$$
This says that for every vertex there to be an edge that is selected</td>
</tr>
<tr>
<td>$$\text{Max } \sum_e x_e$$
$$A.x \leq 1$$
$$x_e \in \{0,1\}$$</td>
<td>$$\text{Min } \sum_e x_e$$
$$A^T y \geq 1$$
$$y_v \in \{0,1\}$$</td>
</tr>
<tr>
<td>x is selection of an edge => $x \in \{0,1\}$
A is an incident matrix of size vxe</td>
<td>y is selection of a vertex.
A is an incident matrix of size exv</td>
</tr>
</table>

This is an IP Duality. Strong duality does not hold for IP duals. These two will have a duality gap.

When will IP duals have strong duality?

Slack Variables :

Consider a problem,

$$\text{Max } c^T.x'$$

Subject to :

$$A'x' \leq b$$

Let A is an mxn matrix (m constraints, n variables)

We add a slack variable to x an make the inequality an equality i.e., $A'x' \leq b$ to

$$Ax = b$$

Where A is $m \times (m+n)$ (we are adding one slack variable per constraint) and x is $(m+n) \times 1$

Example :

$a_1x_1 + a_2x_2 \leq b_1$ becomes $a_1x_1 + a_2x_2 + x_3 \leq b_1$

Here we are adding more variables than the equations. So, many possible solutions exist. We are interested in *Basic Feasible Solutions (BFS)* (or vertices, our solution will lie in one of these vertices) which have m (possibly) non-zero and n zero elements. Now the vector corresponding to non-zero elements is x_B.

BFS is solution to the equation $A_Bx_B = b$ where A_B is $m \times m$ sub-matrix (rest n variables in x are 0)

$$x_B = A_B^{-1}b$$

<u>Example :</u>

$$\text{Maximize } z = x_1 + x_2$$

Subject to :

$$2x_1 + x_2 \leq 2$$

$$x_1 + 2x_2 \leq 2$$

$$x_1, x_2 \geq 0 ;\ x_1 \in \mathbb{Z} ; x_2 \in \mathbb{Z}$$

Now we write $2x_1 + x_2 \leq 2$ as $2x_1 + x_2 + x_3 = 2$ and $x_1 + 2x_2 \leq 2$ as $x_1 + 2x_2 + x_4 = 2$

$$A' = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \quad b = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \quad A = \begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & 2 & 0 & 1 \end{bmatrix}$$

There are ${}^4C_2 \times {}^2C_1 = 6$ possibilities for A_B. columns that we are not picking will have 0 solutions.

$$\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 2 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

We want a sparse vector for x. If we solve $x_B = A_B^{-1}b$ for all the 6 possibilities, corresponding x_B and x are :

$$\begin{bmatrix} 2/3 \\ 2/3 \end{bmatrix} \begin{bmatrix} 2 \\ -2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 2 \\ -2 \end{bmatrix} \begin{bmatrix} 2 \\ 2 \end{bmatrix}$$

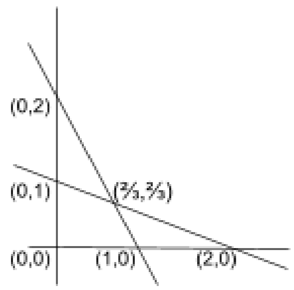
And x are :

$$\begin{bmatrix} 2/3 \\ 2/3 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 2 \\ 0 \\ -2 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 2 \\ -2 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 2 \\ 2 \end{bmatrix}$$

Discarding points with negative x_i we get four vertices :

$$\begin{bmatrix} 2/3 \\ 2/3 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

All these four points we obtained are the vertices of the feasible region :



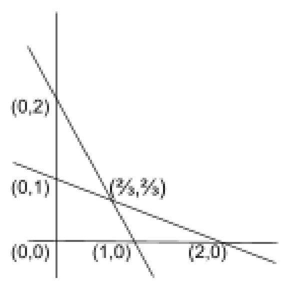
$$A'x' \leq b \implies Ax = b \implies \textit{sparse solution of x (BFS)} \implies \textit{vertices} \implies \textit{search}$$

Here, depending on the no of variables we select no of constraints and we solve them.

Simplex :

Algorithm :

- We start with one of the vertices (BFS) (often by solving the corner) (say $I^{-1}b = x_B$)
- Move to another BFS such that the objective improves.
- Repeat until there is no scope for further improvement
  - Example :



$$c = [1 \quad 1]^T$$

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} : \begin{bmatrix} 0 \\ 0 \end{bmatrix} \longrightarrow \begin{bmatrix} 1 \\ 0 \end{bmatrix} \longrightarrow \begin{bmatrix} 2/3 \\ 2/3 \end{bmatrix}$$

  - $Z = c^T x : 0 \longrightarrow 1 \longrightarrow 4/3$

Reasons for Efficiency :

- Navigation across vertices/BFSs (no search over all vertices)
- Efficient computation of next BFS given the present vertex.
- Being a convex optimization, final optima (on convergence) is the global optima

LP giving IP :

LP gives an IP solution when :

- A is totally Unimodular
- b is integral.

Totally Unimodular :

A matrix is totally unimodular if every submatrix of A has determinant ∈ {-1,0,+1}

Explanation :

$$\text{Max } c^T.x$$

Subject to :

$$Ax \leq b$$

When we solve this we add a slack variable and create a new problem Ax = b

$$x = A^{-1}b$$

X we want is inverse(sub matrix of A).b

$$x_B = A_B^{-1}b$$

$$\begin{aligned} \mathrm{x} &= A^{-1}b \\ &= \frac{1}{\det(\mathrm{A})}\begin{bmatrix} & -1,0,1 & \end{bmatrix}\begin{bmatrix} b \end{bmatrix} \end{aligned}$$

We know that det(A) ∈ {-1,0,+1} and b ∈ ℤ. Therefore, x is integral.

Properties :

Is A^T a TU ?

Is [A,I] a TU?

Matching in BPG :

A matrix in case of BPG is TU :

$$\begin{array}{cc} & |V| \\ |E| & \begin{bmatrix} 1 & & \\ 0 & & \\ 1 & \text{incident matrix (A)} & \\ 0 & & \\ 0 & & \end{bmatrix} \end{array}$$

Proof with induction :

We need to prove :

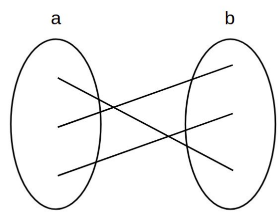
1. We know that every 1x1 sub-matrix of A has a determinant in {-1,0,1}.
2. If $A_{(l-1)x(l-1)}$ is properly TU then A_{lxl} is also TU.

Let us take a sub-matrix of A of size lxl :

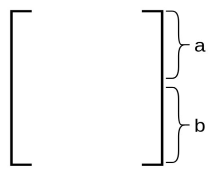
Columns :

We have 3 possibilities :

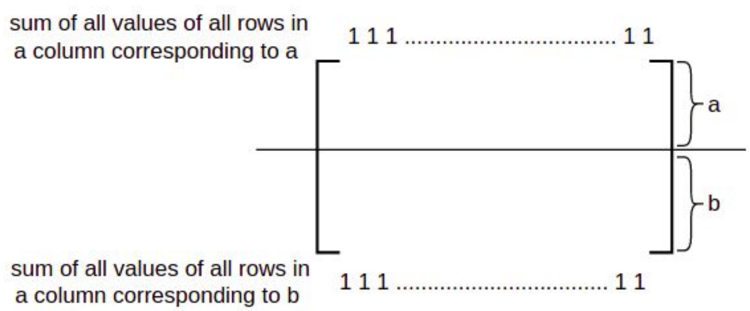
- Let us pick a column with no 1's, all of them are zero.
    - Then we expand along this column and compute determinant along this column. This results in det 0
- Let us pick a column with one 1 an rest all are 0's :
    - Expanding along this column, we look at $A_{(l-1)x(l-1)}$ for this particular 1.
    - This is anyway uni modular. This results in det in {-1,0,1}.
- If there are two 1's we look for another column. If not ?
    - This is a BPG



    - This means that



    - All these edges are from a to b. We can always find a set of rows corresponding to a and rest corresponding to b. If there are two 1's, one will be in the first and the other will be in second.



    - Sum of a set of rows and sum of another set of rows are identical/rows are linearly independent $\Longrightarrow$ determinant = 0
- Therefore, we proved that if all $A_{(l-1)x(l-1)}$ is TU then A_{lxl} is TU

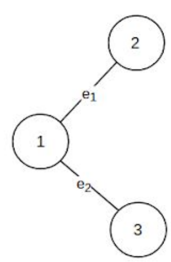
Therefore, in the case of BPG A is TU and we know that b is integral. So, when we solve BPG we get an integral solution.

Example :

Consider two graphs and the corresponding Incident matrices :

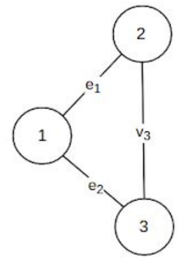
1. BPG :

$$\begin{array}{c c} & \begin{array}{cc} e_1 & e_2 \end{array} \\ \begin{array}{c} v_1 \\ v_2 \\ v_3 \end{array} & \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \end{array}$$



2. Non - BPG :

$$\begin{array}{c|ccc} & e_1 & e_2 & e_3 \\ \hline v_1 & 1 & 1 & 0 \\ v_2 & 1 & 0 & 1 \\ v_3 & 0 & 1 & 1 \end{array}$$



A is not TU in this case. a to a and b to b edge is not possible. non-BPG => Duality gap != 0 (problem is hard)

Example - P:

Primal	Dual
Primal : Max $1^T x$ Subject to : $\begin{bmatrix}1 & 1\\1 & 0\\0 & 1\end{bmatrix}\begin{bmatrix}x_1\\x_2\end{bmatrix} \le 1$	Dual : Min $1^T y$ Subject to : $\begin{bmatrix}1 & 1 & 0\\1 & 0 & 1\end{bmatrix}\begin{bmatrix}y_1\\y_2\\y_3\end{bmatrix} \ge 1$
LP optima is 1 at $[1\ 0]^T$	LP optima is 1 at $[1\ 0\ 0]^T$
LP optimal here is integral, IP optima = LP optima Duality gap = 0	

Example - D:

Primal	Dual
Primal : Max $1^T x$ Subject to : $\begin{bmatrix}1 & 1 & 0\\1 & 0 & 1\\0 & 1 & 1\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix} \le 1$	Dual : Min $1^T y$ Subject to : $\begin{bmatrix}1 & 1 & 0\\1 & 0 & 1\\0 & 1 & 1\end{bmatrix}\begin{bmatrix}y_1\\y_2\\y_3\end{bmatrix} \ge 1$
LP optima is 1 at $[½\ ½\ ½]^T$	LP optima is 2 at $[½\ ½\ ½]^T$
LP optimal here is integral, IP optima = LP optima Duality gap = 1	

A in Max flow-Min cut is TU :

b	1	1	0	0	0		c
A	x_1	x_2	x_3	x_4	x_5		
y_1	1	0	0	0	0		10
y_2	0	1	0	0	0		5
y_3	0	0	1	0	0	≤	15
y_4	0	0	0	1	0	=	5
y_5	0	0	0	0	1	≥	10
y_6	1	0	−1	−1	0		0
y_7	0	1	1	0	−1		0

We are only showing that matrix has a specific property, not for one specific instance of it.

If an IP solution has strong Duality, then A needs to be TU.

https://www.notion.so/Computer-Vision-10a3fe41c59e42c7a5e23928f5322de9

Lagrangian Multiplier :

For instance, let us look into a problem of the form, *Min f(x) subject to g(x) = 0* and let us look into a powerful way to formulate and solve these kind of problems.

Let us look into an optimization problem,

$$\text{Min } x^2 + y^2$$

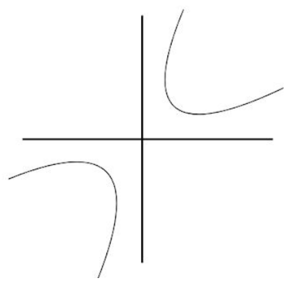
Subject to :

$$xy = 5$$

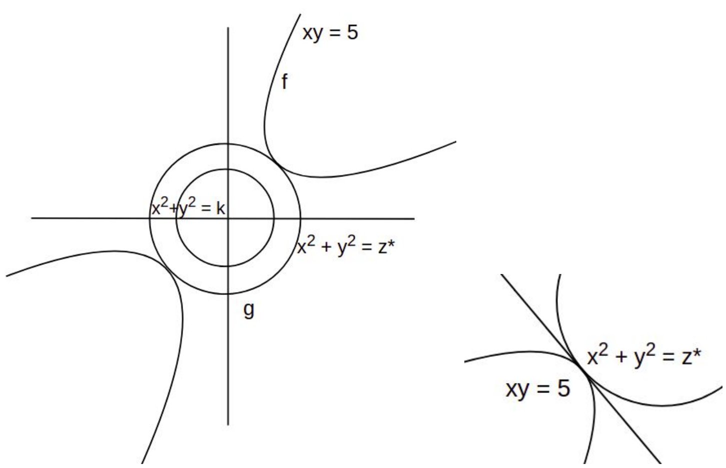
General representation :

Min f(x) subject to g(x) = 0

$$xy = 5 :$$



Optima of $x^2 + y^2$ will be where $xy = 5$ and $x^2 + y^2$ will meet tangentially, they have a common tangent.



We can say that $\mathbf{\nabla f = \lambda \nabla g}$. the scale factor lambda, this scale factor \lambda is because gradients(tangents) have direction but different magnitudes. λ is a Lagrangian multiplier.

Partial derivative with respect to x :

$$2x = \lambda y$$

Partial derivative with respect to y :

$$2y = \lambda x$$

New constraints :

$$2x = \lambda y$$
$$2y = \lambda x$$
$$xy = 5$$

Therefore,

$$\begin{bmatrix} 2 & -\lambda \\ -\lambda & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 0$$

This has a non trivial solution only when det(A) = 0

$$4 - \lambda^2 = 0$$

$$\lambda = \pm 2$$

λ = -2 is not possible, λ = 2 :

$$(x,y) = (\sqrt{5},\sqrt{5}) \text{ or } (-\sqrt{5},-\sqrt{5})$$

Conclusion :

<table>
<tr><td colspan="2">Min f(x) subject to g(x) = 0</td></tr>
<tr><td>Method 1 :
$\nabla f = \lambda \nabla g$
g(x) = 0</td><td>Method 2 :
We create a lagrangian function L(x,y,λ) = f(x,y) - λg(x,y) and equate partial derivatives to 0. i.e.,
$\frac{\partial L}{\partial x} = 0$
$\frac{\partial L}{\partial y} = 0$</td></tr>
</table>

Similarly,

Max f(x,y) subject to g(x) = 0
$L(x,y,\lambda) = f(x,y) - \lambda(g(x,y))$

This gives me

End note : We understood lagrangian multiplier, its connection to lagrangian function and its geometry, how it can be used to solve.

Example :

$$\text{Maximize } x^2y \text{ ; subject to : } x^2+y^2 = 1$$

Solution :

On introducing lagrangian multiple :

We get :

$$L(x,y,\lambda) = x^2y - \lambda(x^2+y^2-1)$$

$\frac{\partial L}{\partial x} = 0$, $\frac{\partial L}{\partial y} = 0$ and $\frac{\partial L}{\partial \lambda} = 0$ leads to :

$$2xy = 2\lambda x \text{ ; } x^2 = 2\lambda y \text{ and } x^2+y^2 = 1$$

We get x,y as $(\pm\sqrt{2/3},\pm\sqrt{1/3})$

Example :

Problem :

Find the maximum possible area of a rectangle subject to the constraint that its perimeter is 20.

Formulation :

$$\text{Maximize } xy \text{ ; subject to : } x+y = 10$$

Solution :

$$L(x,y,\lambda) = xy - \lambda(x+y-10)$$

$\frac{\partial L}{\partial x} = 0, \frac{\partial L}{\partial y} = 0$ and $\frac{\partial L}{\partial \lambda} = 0$ leads to :

$$y = \lambda, x = \lambda \text{ and } x+y=10$$

Therefore, $\lambda = x = y = 5$

Example :

Problem :

Find the points on the circle $x^2+y^2 = 80$, that is closest and farthest from (1,2)

Formulation :

$$\text{Maximize } f(x,y) = (x-1)^2 + (y-2)^2$$

Subject to :

$$x^2+y^2 = 80$$

Solution :

$$L(x,y,\lambda) = (x-1)^2 + (y-2)^2 - \lambda(x^2+y^2 - 80)$$

$\frac{\partial L}{\partial x} = 0, \frac{\partial L}{\partial y} = 0$ and $\frac{\partial L}{\partial \lambda} = 0$ leads to :

$$2(x-1) - \lambda(2x) = 0$$
$$2(y-2) - \lambda(2y) = 0$$
$$x^2+y^2 - 80 = 0$$

From 1 and 2 ;

$$x = 1/1-\lambda \text{ and } y = 2/1-\lambda \text{ i.e., } y = 2x;$$
$$x^2+4x^2 = 80$$

substituting these in 3 gives ;

$$1/1-\lambda = \pm 4. \ (x,y) \text{ are } (\pm 4,\pm 8)$$

Example :

Problem :

Find the maximum and minimum values of $f(x,y) = 81x^2 + y^2$; subject to the constraint $4x^2 + y^2 = 16$

Formulation :

$$\text{Minimize and maximize } 81x^2 + y^2$$

Subject to :

$$4x^2 + y^2 = 16$$

Solution :

$$L(x,y,\lambda) = 81x^2 + y^2 - \lambda(4x^2 + y^2 - 16)$$

$\frac{\partial L}{\partial x} = 0, \frac{\partial L}{\partial y} = 0$ and $\frac{\partial L}{\partial \lambda} = 0$ leads to :

$$81(2x) - \lambda(4(2x)) = 0$$
$$2y - \lambda(2y) = 0$$
$$(4x^2 + y^2 - 16) = 0$$

From constraint 1 :

$$x = 0 \text{ or } \lambda = 81/4$$

From constraint 2 :

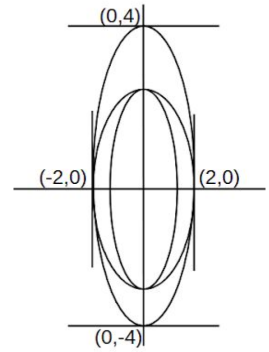
$$y = 0 \text{ or } \lambda = 1$$

From constraint 3 :

To satisfy constraint 3 and x and y together can't be 0. So, possible solutions of (x,y) are :

- $(0,\pm4)$ [when x=0 and $\lambda = 1$]
- $(\pm2,0)$ [when y=0 and $\lambda = 81/4$]

so, minimum among two possibilities $(0,\pm4)$ *and* $(\pm2,0)$ *is at* $(\pm2,0)$. *Optimal solution is 16 at* $(\pm2,0)$ *and maximum is at* $(0,\pm4)$ *with an optimal value of 324*



Example :

Problem :

Find the maximum and minimum values of $f(x,y,z) = 3x^2 + y$; subject to constraints $4x-3y = 10$ and $x^2 + z^2 = 10$

Formulation :

$$\text{Max and min } 3x^2 + y$$

Subject to :

$$4x-3y = 10$$

$$x^2 + z^2 = 10$$

Solution :

On introducing lagrangian multiple i.e., $L(x,y,z,\lambda) = f(x,y,z) - \lambda_1 g_1(x,y) - \lambda_2 g_2(x,y,z)$. We get,

$$L(x,y,z,\lambda) = 3x^2 + y - \lambda_1(4x-3y-10) - \lambda_2(x^2 + z^2-10)$$

$\frac{\partial L}{\partial x} = 0,\ \frac{\partial L}{\partial y} = 0\ ,\ \frac{\partial L}{\partial z} = 0\ ,\ \frac{\partial L}{\partial \lambda_1} = 0$ and $\frac{\partial L}{\partial \lambda_2} = 0$ leads to :

$$3(2x) -\lambda_1(4) - \lambda_2(2x) = 0$$

$$1 - \lambda_1(-3) = 0$$

$$\lambda_2(2z) = 0$$

$$4x-3y-10$$

$$x^2 + z^2-10$$

From constraint 1 :

$$x = (4\lambda_1)/(6)$$

From constraint 2 :

$$3\lambda_1 = -1$$

From constraint 3 :

$$\lambda_2 = 0 \text{ or } z = 0$$

From constraint 4 :

$$4x-3y = 10$$

From constraint 5 :Φ

$$x^2 + z^2 = 10$$

Case 1 ($\lambda_2 = 0$):

$$x = (2\lambda_1)/(3)$$

On substituting constraint 2 [$\lambda_1 = -⅓$] in this we get :

$$x = -2/9$$

Substituting x in constraint 3 :

$$y = -98/27$$

Optimal value of x and y in this case is -94/27 at (-2/9,-98/27)

Case 2 (z= 0):

Therefore, on substituting z=0 in constraint 5 we get :

$$x = \pm\sqrt{10}$$

On substituting $x = \pm\sqrt{10}$ in constraint 4 we get :

$$y = (\pm 4\sqrt{10}-10)/3$$

On substituting these in objective function we conclude that optimal values are $30+((\pm 4\sqrt{10}-10)/3)$ at $(\pm\sqrt{10},(\pm 4\sqrt{10}-10)/3)$

From all the 3 possible values minimum is -94/27 and maximum is $(80+4\sqrt{10})/3$

Lagrange Duality :

Consider a problem ,

$$\text{Min } f(x) \text{ ; subject to } g_i(x) \leq 0 \ \forall\, i \in \{1,2,3,..,n\}$$

$$L(x,\lambda) = f(x) + \sum_{i=1}^{n} \lambda_i g_i(x) \ \ ; \lambda \geq 0$$

$$L(x,\lambda) = f(x) + \lambda^T g$$

In lagrangian function, when x is feasible :

$$\text{Max } L\big|_x \text{ is } f(x)$$

Explanation :

λ's are positive and when they are

- feasible $\lambda^T g \leq 0$; therefore, max (L) is f(x)
- not feasible ; max (L) is infinite

Primal :

$$\boxed{\text{Min } (\text{Max } L(x,\lambda)\big|_{\lambda\geq 0})\big|_x}$$

Corresponding dual problem is :

$$\boxed{\text{Max } (\text{Min } L(x,\lambda)\big|_x)\big|_{\lambda\geq 0}}$$

Min-max theory tells us that :

$$\underset{x}{\text{Max}} \ (\underset{y}{\text{Min}} \ \Phi(x,y)\ ) \leq \ \underset{y}{\text{Min}} \ (\underset{x}{\text{Max}} \ \Phi(x,y)\ )$$

Therefore, lagrangian duality is weak.

Example :

Consider LP in standard form :

$$p^* = \text{Maximize } c \cdot x$$

$$\text{Subject to } A.x \leq b$$

With no additional constraint on x

$$d^* = \text{Minimize } b \cdot y$$

$$\text{Subject to } A^T y = c,\ y \geq 0.$$

Consider the LP in standard inequality form :

$$p^* = \text{Maximize } c \cdot x$$

Lagrangian function is

$$L(x,\lambda) = c^T x + \lambda^T(b-Ax)$$

We can see that this function is $\geq c^T x$ when $\lambda \geq 0$. Let us define the dual function $g(\lambda)$ as

$$g(\lambda) = \max L(x,\lambda) \big|_x \;\geq\; p^*$$

We are interested in best(tightest) upper bound on $g(\lambda)$

$$d^* = \text{Min}(g(\lambda)) \big|_{\lambda\geq 0} = \text{Min}(\ \text{Max}(c^T x + \lambda^T(b-Ax) \big|_x)\big|_{\lambda\geq 0}$$

$$d^* = \text{Min}(\ c^T x + \text{Max}(\lambda^T(b-Ax) \big|_x)\big|_{\lambda\geq 0}$$

We know that x is not constrained to be positive. If c^T - A has non zero entries then maximum over all x is infinity and is a useless upper bound. We should only consider the case when $A^T\lambda = c$. This leads to a dual problem as :

$$d^* = \text{Min}(b^T\lambda) \;\; ; \;\text{Such that} : A^T\lambda = c \text{ and } \lambda \geq 0$$

Optimality : unconstrained optimization :

First Order Necessary Conditions for optimality :

- x^* is a local optimum if it is a stationary point

$$\nabla_x J(x^*) = 0$$

Second Order Sufficient Conditions for optimality :

- If

$$\nabla_x J(x^*) = 0 \text{ and } \nabla_x^{\,2} J(x^*) \succ 0$$

  x^* is a strict local minimum
- If

$$\nabla_x J(x^*) = 0 \text{ and } \nabla_x^{\,2} J(x^*) \prec 0$$

  x^* is a strict local maximum

Lagrangian Multiplier for Inequality Constraints :

$$\text{Min } f(x) \; ; \text{ subject to } g_i(x) \leq 0 \;\forall i \in \{1,2,3,..,m\}$$

Create lagrangian :

Now we add a slack variable ;

New problem is

$$\text{Min } f(x) \; ; \text{ subject to } g_i(x)+s_i^{\,2} = 0 \;\forall i \in \{1,2,3,..,m\}$$

We can now write lagrangian :

$$L(x,\lambda,s) = f(x) + \sum_{i=1}^{n} \lambda_i (g_i(x)+s_i^{\,2}) \;\; ; \lambda \geq 0$$

$\frac{\partial L}{\partial x} = 0,\ \frac{\partial L}{\partial s_i} = 0$ and $\frac{\partial L}{\partial \lambda} = 0$ leads to :

$$\nabla f + \sum_{i=1}^{n} \lambda_i \nabla g_i = 0$$

$$g_i(x)+s_i^{\,2} = 0 \text{ or } g_i(x) \leq 0$$

$$\lambda_i s_i = 0 \;/\; \lambda_i g_i = 0$$

Constraint 1 is optimal ; Constraint 2 is primal feasibility constraint ; Constraint 3 is complementary slackness : either dual variable is 0 or the constraint is tight.

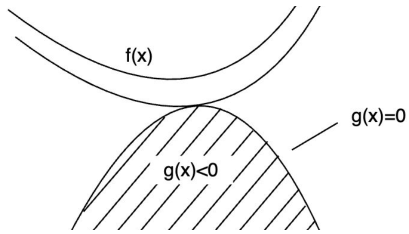
Dual feasibility constraint :

$$\lambda \geq 0$$

In general all these constraints, along with equality constraints are KKT conditions.

Situation 1 :

In the equality case we did not have a constraint on the sign of lagrangian multiplier.



Optimal point where these both have common tangent.

$$\nabla f + \lambda_i \nabla g_i = 0$$

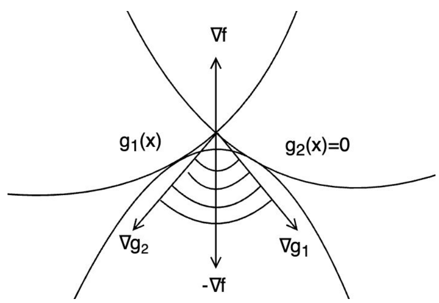
Therefore,

$$\lambda = - \frac{\nabla f}{\nabla g}$$

$\lambda \geq 0$; because we know that ∇f and ∇g increase in opposite direction.

Situation 2 :

Point of intersection of g_1 and g_2 is optimal point and :



$-\nabla f = \sum_{i=1}^{n} \lambda_i \nabla g$ => $-\nabla f$ is in the cone defined by the constraints at this point.

Conclusion :

$$\text{Max } f(x)$$

Subject to :

$$g_j(x) \leq 0 \ \forall j \in \{1,2,3,..,m\}$$
$$h_j(x) = 0 \ \forall j \in \{1,2,3,..,l\}$$

Lagrangian :

$$L(x,\lambda,s) = f(x) + \sum_{i=1}^{n} \lambda_i (g_i(x) + s_i^2) + \sum_{i=1}^{n} \mu_j h_j(x)$$

Then at x* :

$$-\nabla f(x) = \sum_{i=1}^{n} \lambda_i \nabla (g_i(x)) + \sum_{i=1}^{n} \mu_j \nabla h_j(x)$$

$$g_i(x^*) \leq 0 \text{ and } h_j(x^*) = 0$$

$$\lambda_j g_j(x^*) = 0$$

$$\lambda_j >= 0 \text{ and } \mu_j \in R$$

Derivative of f is linear combination of derivatives of g's (with positive coefficients) and h'. $g_i(x^*) \leq 0$ and $h_j(x^*) = 0$ are feasibility constraints, optima has to satisfy this. Constraint 3 is complementary slackness, this says that either lambda is 0 or the constraint is tight. Constraint 4 is

Note :

- If we do $L(x,\lambda,s) = f(x) - \sum_{i=1}^{n} \lambda_i(g_i(x)+s_i^2) + \sum_{i=1}^{n} \mu_j h_j(x)$ then either $g_i(x) >= 0$
- These KKT conditions ain't there just to help us in solving problems or to define Lagrangian Duality. They tell us a lot about optimization problems, associated constraints,.. We need to make note of how these constraints are related to LP.

Example :

$$\text{Min } x_1^2 - 4x_1 + x_2^2 - 6x_2$$

Subject to :

$$x_1+x_2 \leq 3 \qquad (g_1)$$
$$-2x_1+x_2 \leq 2 \qquad (g_2)$$
$$x_1+x_2 >= 0 \qquad (g_3,g_4)$$

Lagrangian :

KKT Conditions yield :

$\frac{\partial L}{\partial x_1}, \frac{\partial L}{\partial x_2}, \frac{\partial L}{\partial \lambda_1}, \frac{\partial L}{\partial \lambda_2}, \frac{\partial L}{\partial s_1}, \frac{\partial L}{\partial s_2}$ (complementary slack) and $\lambda_i \geq 0$ yeilds :

$$2x_1 - 4 + \lambda_1 - 2\lambda_2 = 0$$
$$2x_2 - 6 + \lambda_1 + \lambda_2 = 0$$
$$x_1+x_2 \leq 3$$
$$-2x_1+x_2 \leq 2$$
$$\lambda_1(x_1+x_2-3) = 0$$
$$\lambda_2(-2x_1+x_2-2) = 0$$

Constraint 3 gives :

$$\lambda_1 = 0 \text{ or } x_1+x_2 = 3$$

Similar for constraint 4 ; on solving we get $x_1^* = 1$, $x_2^* = 2$; $\lambda_1 = 2$ and $\lambda_2 = 0$.

Note :

We did not consider g_3 and g_4 because this uselessly brings in $x_1 = 0$ or $\lambda_3 = 0$... we anyway get $\lambda_3 = 0$.

Example :

$$\text{Min } -x_2$$

subject to:

$$x_1^2 + x_2^2 - 4 \leq 0$$
$$-x_1^2 + x_2^2 \leq 0$$

KKT conditions lead to :

$\frac{\partial L}{\partial x_1}, \frac{\partial L}{\partial x_2}, \frac{\partial L}{\partial \lambda_1}, \frac{\partial L}{\partial \lambda_2}, \frac{\partial L}{\partial s_1}, \frac{\partial L}{\partial s_2}$ (complementary slack) and $\lambda_i \geq 0$ yeilds :

$$x_1\lambda_1 - x_2\lambda_2 = 0$$
$$-1 + 2x_2\lambda_1 + \lambda_2 = 0$$

$$x_1^2 + x_2^2 \leq 4$$
$$-x_1^2 + x_2^2 \leq 0$$
$$\lambda_1(x_1^2 + x_2^2 - 4) = 0$$
$$\lambda_2(-x_1^2 + x_2^2) = 0$$
$$\lambda_1,\ \lambda_2 \geq 0$$

$x_1 = 0$ and $x_2 = 0$ satisfy these equations. This is not the optimal value.

Sufficiency :

- KKT conditions are only "necessary" for optimality, in general.
- These conditions are sufficient for convex optimization problems.

Example - KKT for LP :

$$\text{Max } x_1+x_2$$

Subject to :

$$x_1 + 2x_2 \leq 4$$
$$2x_1 + x_2 \leq 6$$
$$x_1, x_2 \geq 0$$

KKT for LP :

$$f(x) = x_1+x_2$$
$$g_1(x) = x_1 + 2x_2 - 4 \leq 0$$
$$g_2(x) = 2x_1 + x_2 - 6 \leq 0$$
$$g_3(x) = -x_1 \leq 0$$
$$g_4(x) = -x_2 \leq 0$$

$$\nabla_f = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \nabla_{g_1} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \quad \nabla_{g_2} = \begin{bmatrix} 2 \\ 1 \end{bmatrix} \quad \nabla_{g_3} = \begin{bmatrix} -1 \\ 0 \end{bmatrix} \quad \nabla_{g_4} = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

We know that partial derivative of f is linear combination of partial derivatives of g's

$$\begin{bmatrix} 1 \\ 1 \end{bmatrix} = \lambda_1 \cdot \begin{bmatrix} 1 \\ 2 \end{bmatrix} + \lambda_2 \begin{bmatrix} 2 \\ 1 \end{bmatrix} + \lambda_3 \cdot \begin{bmatrix} -1 \\ 0 \end{bmatrix} + \lambda_4 \cdot \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

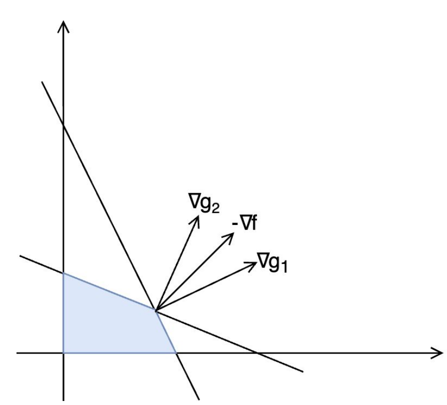
On expanding this we get :

$$\lambda_1 + 2\lambda_2 - \lambda_3 = 1$$
$$2\lambda_1 + \lambda_2 - \lambda_4 = 1$$

Or

Since λ_3 and $\lambda_4 \geq 0$ makes them :

$$\lambda_1 + 2\lambda_2 \geq 1$$
$$2\lambda_1 + \lambda_2 \geq 1$$



This is $A^T y \geq c$

Or

$$\lambda_1 = \lambda_2 = ⅓\ ;\ \lambda_3 = \lambda_4 = 0$$

$\lambda_3 = \lambda_4 = 0$ because constraints are tight.

Example :

Maximum area given perimeter = 100

Max xy

Subject to :

x + y = 50

x ≥ 0 ; y ≥ 0

$$\nabla_f = \begin{bmatrix} y \\ x \end{bmatrix} \quad \nabla_h = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \nabla_{g_1} = \begin{bmatrix} -1 \\ 0 \end{bmatrix} \quad \nabla_{g_2} = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

$$\begin{bmatrix} y \\ x \end{bmatrix} = \lambda_1 \cdot \begin{bmatrix} -1 \\ 0 \end{bmatrix} + \lambda_2 \cdot \begin{bmatrix} 0 \\ -1 \end{bmatrix} + \mu \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

From this we get 2 equations and solving them further we get $\lambda_1 = \lambda_2 = 0$ and $\mu = 25$. Inequalities disappeared because they are not tight.

Lagrange Duality :

Max f(x)

Subject to :

$g_j(x) \leq 0 \ \forall j \in \{1,2,3,..,m\}$

$h_j(x) = 0 \ \forall j \in \{1,2,3,..,l\}$

is equivalent to the optimization problem,

Primal :	Dual :
$$\min\left(\max\left(f(x) + \sum_j \mu_j g_j(x) + \sum_i \lambda_i h_i(x)\right)\Big\rvert_{\mu\geq 0,\lambda}\right)\Big\rvert_x$$ $$\min(\max(L(\text{x},\mu \geq 0, \lambda))\rvert_{\mu\geq 0,\lambda})$$	$$\max\left(\min\left(f(x) + \sum_j \mu_j g_j(x) + \sum_i \lambda_i h_i(x)\right)\Big\rvert_{\text{x}}\right)\Big\rvert_{\mu\geq 0,\lambda}$$ $$\max(\min(L(\text{x},\mu \geq 0, \lambda))\rvert_{\text{x}})\rvert_{\mu\geq 0,\lambda}$$
$$\max(L(\text{x},\mu \geq 0, \lambda)) = \begin{cases} f(x) & \text{when x is feasible} \\ \infty & \text{otherwise} \end{cases}$$	
Dual lower bound of primal (Weak Duality) : From max-min inequality theorem : $$\max(\min(L(.))\rvert_{\text{x}})\rvert_{\mu\geq 0,\lambda} \leq \min(\max(L(.))\rvert_{\mu\geq 0,\lambda})\rvert_{\text{x}}$$ This inequality becomes equality when optimization is convex optimization (SVM) i.e., strong duality exists for all convex optimization problems.	

One advantage of lagrangian multiplier is : either it is 0 or non-zero

SVM :

SVM - to separate two different classes is at least

Primal Problem :

$$\text{Minimize } \tfrac{1}{2}\, w.w \,\rvert_{w,b}$$

Subject to :

$$(w.x_j+b)y_j \geq 1 \quad \forall j$$

w - weight of the features

y_j is class label

Lagrangian :

$$L(\mathrm{w},\mathrm{b},\alpha) \;=\; \frac{1}{2}\mathrm{w}.\mathrm{w} \;-\; \sum_j \alpha_j\left[(\mathrm{w}.\mathrm{x}_j + \mathrm{b})\mathrm{y}_j - 1\right] \quad \alpha_j \geq 0,\, \forall j$$

$$\alpha \;-\; \text{weights on training points}$$

Complementary Slackness :

$\alpha_j \geq 0$ constraint is effective

$(w.x_j+b)y_j = 1$

Point j is a support vector

Or

$\alpha_j = 0$, so point j is not a support vector

x_j is not variable. w and b are variables we have to find to optimize subjective function. Here, $g(x) \geq 0$ so we use a negative sign before α*g(x) and $\alpha \geq 0$.

Dual derivation :

$$\max\left(\min\left(L(\mathrm{w},\mathrm{b},\alpha)\right)_{\mathrm{w},\mathrm{b}}\right)_\alpha \;=\; \max\left(\min\left(\frac{1}{2}\mathrm{w}.\mathrm{w} \;-\; \sum_j \alpha_j\left[(\mathrm{w}.\mathrm{x}_j + \mathrm{b})\mathrm{y}_j - 1\right]\right)_{\mathrm{w},\mathrm{b}}\right)_\alpha \quad \alpha_j \geq 0,\, \forall j$$

$$\frac{\partial L}{\partial \mathrm{w}} = 0 \implies \mathrm{w} = \sum_j \alpha_j \mathrm{y}_j \mathrm{x}_j$$

$$\frac{\partial L}{\partial \mathrm{b}} = 0 \implies \sum_j \alpha_j \mathrm{y}_j \mathrm{x}_j \;=\; 0$$

(if we can solve for s(dual problem) then we have a solution for w (primal problem))

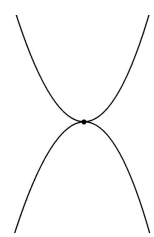
Therefore,

Dual problem is :

$$\mathrm{maximize}_\alpha \sum_i \alpha_i \;-\; \frac{1}{2}\sum_{i,j} \alpha_i\alpha_j \mathrm{y}_i\mathrm{y}_j\mathrm{x}_i\mathrm{x}_j$$

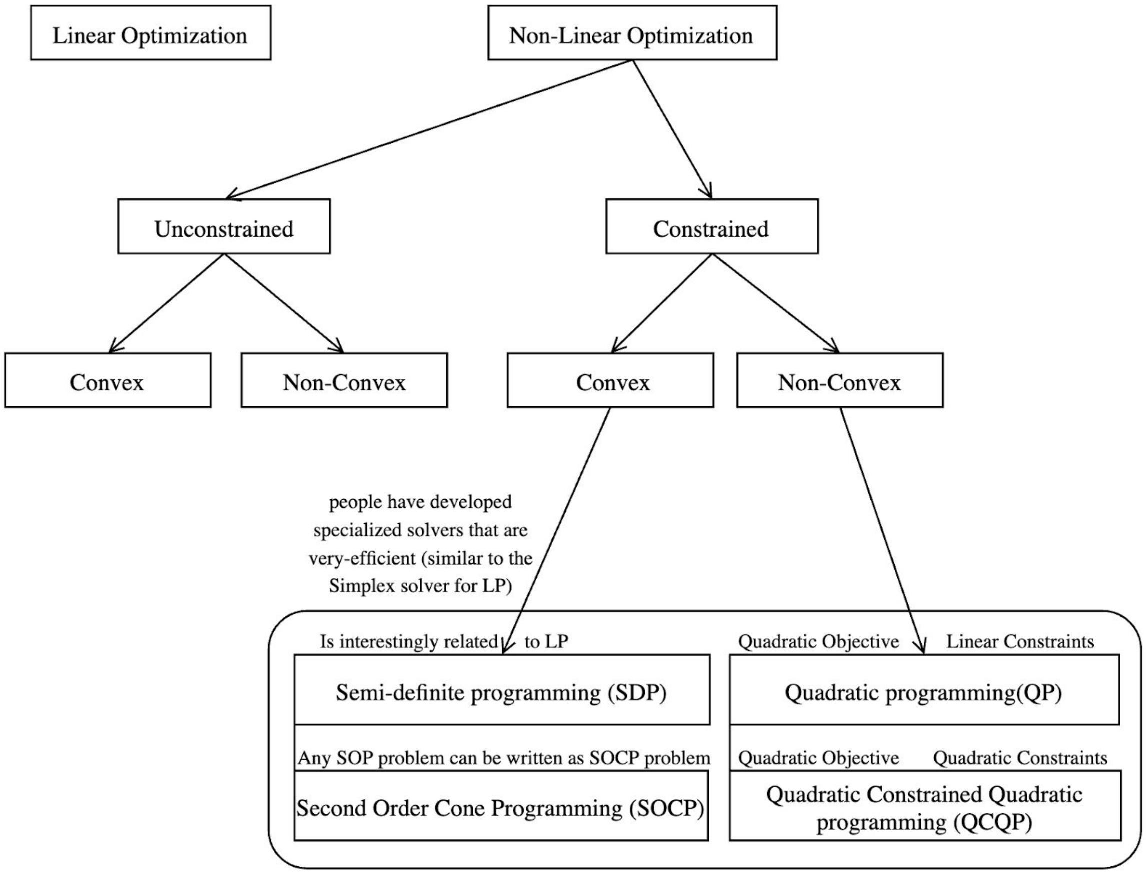
$$\sum_i \alpha_i \mathrm{y}_i \;= 0$$

$$\alpha_i \;\geq 0$$



SVM is convex, we get the same solution with both primal and dual.

Classification Chart :



Knowing about the class to which the problem belongs to is necessary because they help us solve as we know some standard methods if a problem belongs to a particular class.